XIII. Magyar Számítógépes Nyelvészeti Konferencia

MSZNY 2017

Szerkesztette:

Vincze Veronika

Szeged, 2017. január 26-27.
http://rgai.inf.u-szeged.hu/mszny2017

ISBN: 978-963-306-518-1

Szerkesztette: Vincze Veronika
vinczev@inf.u-szeged.hu

Felelős kiadó: Szegedi Tudományegyetem, Informatikai Intézet
6720 Szeged, Árpád tér 2.

Nyomtatta: JATEPress
6722 Szeged, Petőfi Sándor sugárút 30–34.

Szeged, 2017. január

Előszó

2017. január 26-27-én tizenharmadik alkalommal rendezzük meg Szegeden a Magyar Számítógépes Nyelvészeti Konferenciát. A konferencia fő célkitűzése a kezdetek óta állandó: a nyelv- és beszédtechnológia területén végzett legújabb, illetve folyamatban levő kutatások eredményeinek ismertetése és megvitatása, ezen felül lehetőség nyílik különféle hallgatói projektek, illetve ipari alkalmazások bemutatására is.

Örömet jelent számunkra, hogy a hagyományokat követve a konferencia idén is nagyfokú érdeklődést váltott ki az ország nyelv- és beszédtechnológiai szakembereinek körében. Idén először teljes munkák beküldésével lehetett jelentkezni a konferenciára, melyek alapos elbírálása után döntött a programbizottság a cikkek elfogadásáról. A nagy számban beérkezett tudományos cikkek közül idén a programbizottság 26 előadást, 5 poszter-, illetve 4 laptopos bemutatót fogadott el. A tavalyi évhez hasonlóan, egyes témákat mind az előadások, mind pedig a laptopos bemutatók között is megtalálunk, ezzel is lehetőséget adva a kutatási témák minél szélesebb körű bemutatására. A programban a magyar számítógépes nyelvészet rendkívül széles skálájáról találhatunk előadásokat a számítógépes morfológiától kezdve a beszédtechnológián át a szentimentelemzésig. Mindemellett a magyar nyelvtechnológiai műhelyek együttműködésében megvalósult, egy egységes magyar előfeldolgozó láncot kifejlesztő e-magyar.hu projekt eredményei bemutatásának is külön szekciót szentelünk.

Örömünkre szolgál az a tény is, hogy Labádi Gergely, a Szegedi Tudományegyetem Magyar Irodalmi Tanszékének docense elfogadta meghívásunkat, és a digitális bölcsészetről szóló plenáris előadása is gyarapítja a konferencia résztvevőinek szakmai ismereteit.

Az idei évben – reményeink szerint új hagyományt teremtve – szeretnénk különdíjjal jutalmazni a konferencia legjobb cikkét, mely a legkiemelkedőbb eredményekkel járul hozzá a magyarországi nyelv- és beszédtechnológiai kutatásokhoz. A díj anyagi hátterét az MTA Nyelvtudományi Intézete biztosítja, amiért ezúton is hálás köszönetet mondunk. Továbbá szeretnénk megköszönni a programbizottság és a szervezőbizottság minden tagjának áldozatos munkájukat, nélkülük nem jöhetett volna létre a konferencia.

Csirik János
Farkas Richárd
Simon Eszter
Vincze Veronika

Szeged, 2017. január

Tartalomjegyzék

I. Információkinyerés

II. e-magyar előadások

III. Beszédtechnológia

IV. Szentimentelemzés

V. Többnyelvűség

VI. Poszterek

VII. Laptopos bemutatók

VIII. Angol nyelvű absztraktok

I. Információkinyerés

Ablak által világosan – Vonzatkeret-egyértelműsítés az igekötők és az infinitívuszi vonzatok segítségével

Vadász Noémi[1,3], Kalivoda Ágnes[1], Indig Balázs[2,3]

[1]Pázmány Péter Katolikus Egyetem, Bölcsészet- és Társadalomtudományi Kar
[2]Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
[3]MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport
{vadasz.noemi,kalivoda.agnes,indig.balazs}@itk.ppke.hu

Kivonat A mondat elemei közötti viszonyok feltárásakor a vonzatkeretek mielőbbi egyértelműsítésére törekszünk. A vonzatkeret-egyértelműsítésben az igekötők és az infinitívuszi vonzatok kiemelt szerepet kapnak. Ha ezek az elemek az igétől jobbra helyezkednek el, az a balról jobbra történő elemzés során problémát jelenthet. Cikkünkben az ige–igekötő, ige–infinitívuszi vonzat távolságok korpuszokon történő kimérése után egy megoldást kínálunk erre a problémára. Az általunk készített *VFrame* keresőeljárás az igékhez esetlegesen tartozó igekötőket és infinitívuszi vonzatokat egységesen, egy eljárásban kezeli, így a vonzatkeret-egyértelműsítés azonnal elvégezhető.

1. Bevezetés

Az AnaGramma elemzőrendszer [1] egy pszicholingvisztikai indíttatású nyelvelemző modell [2], amely az emberi mondatmegértés mintájára balról jobbra és szavanként elemez. A rendszer működésének három alapvető eleme a *kereslet-kínálat* elvű keretrendszer [3], a *tározó* és az *ablak* (a részleteket lásd 1.1). Az elemző kimenete egy függőségi éleket tartalmazó gráf.

A kereslet-kínálat elvű keretrendszer azt jelenti, hogy az elemző látóterébe kerülő tokenek más – korábbi vagy éppen később érkező – elemek számára kínálatként szerepelhetnek, ugyanakkor saját keresleteik is lehetnek (pl. az ige keresi vonzatait). A keresleteket *keresőeljárások* valósítják meg, amelyek különböző megszorításokat tartalmaznak pl. a keresés irányára. Emellett olyan egyéb információkat is hordozhatnak, amelyek a mondat elemzése során később lehetnek szükségesek. A keresőeljárások egyik legfontosabb funkciója a *találati függvény*, amely utasítja az elemzőt, hogy mit tegyen, ha a keresés a keresett elem megtalálásával vagy sikertelenül fejeződött be. Az eljárás az eredménye alapján (talált/nem talált) az elemzés következő lépésében újabb keresőeljárást indíthat.

Az elemzés során az adott pillanatig még be nem kötött éleket, tehát az eddigi kínálatokat (az éppen elemzett tokent megelőző tokeneket, azok morfológiai elemzését és az esetleges kész szerkezeteket), a keresleteket, valamint a kész részszerkezeteket egy rövidtávú munkamemória [4], az ún. tározó tartalmazza. Az

elemzés bármely pillanatában lekérdezhető a tározó tartalma a keresletek számára. A kereslet és kínálat találkozásakor az elemek jegyeinek unifikálódásával jön létre közöttük kapcsolat.

Az elemzés során a függőségi élek bekötéséhez a tározón kívül rendelkezésre áll egy, az aktuálisan elemzett szótól jobbra tekintő, két token méretű elemzési ablak is. Ebben az ablakban az éppen elemzett token bizonyos keresleteit kielégítő kínálatok is előfordulhatnak, amelyek befolyásolhatják az elemzést.

1.1. Az ablak és a tározó

Az AnaGramma az emberi szövegfeldolgozást modellálja. Ehhez a *Sausage Machine* kétfázisú mondatfeldolgozását [5] veszi alapul. Az első fázis a *Preliminary Phrase Packager* (PPP), amely a szöveges bemenet szócsoportjaiból frázisokat „csomagol". A második fázisban (*Sentence Structure Supervisor*) a „csomagok" további nemterminális csomópontok hozzáadásával megkapják a szerepüket a mondatban[1].

A PPP fázis eredetileg egy kb. hat szó méretű ablakban dolgozik (angol szöveggel). Agglutináló jellege miatt a magyar nyelvre célszerűbb egy kisebb ablak alkalmazása, így a PPP fázist az AnaGramma keretében egy három token méretű elemzési ablak modellálja[2].

A Sausage Machine alkalmazása az AnaGramma modellben azt jelenti, hogy a balról jobbra szavanként történő feldolgozás egy lépésében az elemző az – egyelőre három token méretűnek meghatározott – ablak bal szélén lévő elemmel foglalkozik, elindítja az összes, az adott token morfológiai elemzése által indítandó folyamatot, megvizsgálja, hogy a token mint kínálat kielégíthet-e egy keresletet a tározóban. Az ablakban található további tokenek – az elsődleges morfológiai információjukkal együtt – módosíthatják az aktuális elemzési lépéseket (pl. jelentős szerepük van a szófaji egyértelműsítésben is).

1.2. Vonzatkeret-egyértelműsítés az ablakban

Amikor az elemző egy igei elemet talál – legyen az finit vagy infinit ige, melléknévi vagy határozói igenév, amelynek vonzatkerete van és lehet igekötője –, többféle lehetséges vonzatkeret is felmerülhet. Az elemzés során a vonzatkeret-egyértelműsítésnek minél előbb meg kell történnie ahhoz, hogy a megfelelő keresőeljárások elindulhassanak. Az olyan, saját vonzatkerettel rendelkező igei elemek esetében, amelyeknek lehet igekötőjük, a balról jobbra történő elemzés

[1] Az elemzési ablak fontosságát jól mutatja, hogy nehezebben dolgozzuk fel a hírcsatornákon a képernyő alján végigfutó hírszalagot, ahol nem tudunk előretekinteni (lásd: http://users.itk.ppke.hu/∼yanzigy/olvaso/).

[2] Az ablaknak flexibilisnek is kell lennie, amely azt jelenti, hogy az elemző az elemzési ablakban „átugorja" az aktuális állapota számára nem releváns elemeket. Ezek jellemzően olyan rövid, nem tartalmas funkciószavak, amelyek nem játszanak szerepet a PPP fázisban. Az ablak megfelelő méretének és a flexibilitásának alátámasztásához szemmozgáskövetővel végzett kísérletek eredményei szükségesek, ám magyarra – eddig – ilyen megközelítésű kísérletről nem tudunk.

során felmerül az igei elemtől jobbra elhelyezkedő igekötő és infinitívuszi vonzat problémája. Ekkor az igekötő és az infinitívuszi vonzat mint a vonzatkeret-egyértelműsítésben fontos szerepet játszó elem „később" kerül be az elemzés folyamatába. Az ablak és a tározó, valamint a keresőeljárások segítségével az igék és az esetlegesen hozzájuk tartozó igekötők és infinitívuszi vonzatok egységesen kezelhetők.

1.3. A korpuszok

Méréseinket három különböző korpuszon végeztük. Az általunk készített InfoRádió korpusz rövid politikai és gazdasági híreket tartalmaz az InfoRádió hírportálról. A rövid hírek egy címből és egy két-három mondatos hírből állnak. A szerkesztett szöveg egyfajta „ideális" bemenetként szolgál az elemzőmodellünk számára. A korpusz 54996 hírből, 135587 mondatból és 1953419 tokenből áll.

A Magyar Nemzeti Szövegtár (MNSZ2) [6] v.2.0.3 verzióján is végeztünk méréseket, amely 785 millió tokent tartalmaz (írásjelek nélkül). A korpusz sokféle forrásból épül fel, közöttük fórumhozzászólásokból és beszélgetések leirataiból. Az MNSZ2 kontrasztot képez az InfoRádió korpusszal, hiszen ez utóbbi egy műfajból származó, szerkesztett szöveget tartalmaz.

A Pázmány Korpusz [7] 1,2 milliárd tokenből áll, amit több, mint 30000 weboldalról gyűjtöttek. Megkülönbözteti a főkorpusz (szerkesztett) és a kommentkorpusz (szerkesztetlen) szövegrészeket, amelyeket mi egyben kezeltünk.

2. Korpuszmérések

Két korpuszelemzést végeztünk. Az egyikkel a finit ige és a tőle jobbra álló igekötő lehetséges távolságát mértük ki, a másikkal pedig az infinitívusznak és jobbra kihelyezett igekötőjének a távolságát. Módszerünk lényege egy pozíció szerinti összehasonlítás, amelyben 0 pozíciónak az igét tekintjük, és ehhez képest határozzuk meg az igekötő helyét (+1 pozíció tehát pl. *látta meg*).

Mindhárom korpusz tartalmaz hibás annotációkat, emiatt sok rossz találatot is kaptunk. Ezeknek az automatikus szűrésére egy több mint 27 ezer igekötős igelemmát tartalmazó (manuálisan ellenőrzött) listát [8] használtunk fel. Ezzel a listával átszűrtünk minden korpuszbéli adatot, és csak azt az igekötő–ige párt fogadtuk el, amely a lista alapján létező kombináció. Ezzel állapítottuk meg azt is, hogy a gyűjtött anyagban az adott igekötő a finit vagy infinit igéhez tartozike (esetleg tartozhat-e elvben mindkettőhöz). Így vesztettünk néhány egyébként releváns találatot olyan neologizmusok esetében, amelyek nem szerepeltek a listában, de a módszerrel a hibás találatok jelentős részét hatékonyan, automatikusan ki lehetett szűrni.

2.1. Finit igék, infinitívuszok és igekötőik

Az igekötő mellett az infinitívuszi vonzatnak is van vonzategyértelműsítő szerepe, és mint ilyen, megerősítheti vagy kizárhatja bizonyos igekötők főigéhez való

kapcsolhatóságát. Öt igeosztályba sorolhatjuk az igéket aszerint, hogy igekötő nélküli, illetve igekötős vonzatkeretükben szerepelhet-e infinitívuszi vonzat. Az 1. táblázat foglalja össze az öt igeosztály tulajdonságait.

	igeosztály	példa		
	PreV vagy INF vonzat	tő	PreV	INF
(a)	nincs PreV, nincs INF	*villog*	X	X
(b)	nincs INF	*esik*	el, le...	X
(c)	nincs PreV	*kell*	X	?
(d)	PreV és INF kölcsönösen kizárják egymást	*tud*	le, meg...	?
(e)	INF bizonyos PreV-vel	*megy*	ki, el...	?

1. táblázat. Öt igeosztály az igéknek az igekötővel és az infinitívuszi vonzattal való kombinálhatósága alapján (X: nincs elfogadott infinitívuszi vonzat vagy igekötő az igével kombinálva, ?: az igének lehet, hogy van infinitívuszi vonzata)

A balról jobbra történő elemzés során az ige elemzésének pillanatában az ideális az, ha rendelkezésünkre áll minden vonzatkeret-egyértelműsítő információ. Megvizsgáltunk minden lehetséges kombinációt, amely az igekötő (PreV), finit ige (FIN) és az infinitívusz (INF) egymáshoz viszonyított sorrendjéből jön létre. Az eredményeket a 2. táblázat foglalja össze.

PreV – **FIN** – INF	**meg** sem **próbálták** csökkenteni
PreV – FIN – **INF**	**le** is akartam **fényképezni**
FIN – **PreV** – INF	**szűnjön meg** létezni
FIN – **PreV** – **INF**	sikerült két példányt **el** is **ejtenie**
INF – **FIN** – **PreV**	csodálni **járok vissza**
INF – FIN – **PreV**	**rohannia** kell **vissza**
FIN – INF – **PreV**	-
FIN – **INF** – **PreV**	kellett egészben új állami rendet [...] **építeni fel**
INF – **PreV** – **FIN**	kártyázni **le** ne **ülj**
INF – **PreV** – FIN	**feledni el** nem tudlak
PreV – INF – **FIN**	-
PreV – **INF** – FIN	**el** nem **utasítani** kegyeskedjék

2. táblázat. A finit ige (FIN), az infinitívuszi vonzat (INF) és valamelyik igekötőjének (PreV) egymáshoz viszonyított lehetséges sorrendjei példákkal (vastag betűvel az összetartozó párokat jelöltük)

A leggyakoribb szerkezet az olyan PreV–FIN–INF, amelynél az igekötő az infinitívuszhoz tartozik. Szintén gyakori a FIN–PreV–INF sorrend, ekkor az igekötő legtöbbször a finit igéhez tartozik. Ez a szórend jellemzi a non-neutrális (ezen belül főként a felszólító, tagadó) mondatokat. Az INF–FIN–PreV szerkezetnél az a tendencia figyelhető meg, hogy maga az infinitívusz áll fókuszpozícióban, és ez mozdítja el az igekötőt a preverbális pozícióból. A további három lehetséges

kombinációra is találtunk példát a korpusz mondatai között (lásd a 2. táblázat utolsó három sora), ezek a szerkezetek azonban ritkák.

2.2. A finit ige és a tőle jobbra elhelyezkedő igekötő

Bár a finit ige utáni mondatszakaszban a fő összetevők sorrendje szabad [9], az MNSZ2-n végzett mérések [8] azt mutatták ki, hogy a posztverbális igekötők az esetek 99%-ában +1 vagy +2 pozíciót foglalnak el. Az InfoRádió korpuszban ez az érték 100%, vagyis nem található benne olyan példa, ahol egynél több szó áll a finit ige és annak posztverbális igekötője között. A különbség azzal magyarázható, hogy az InfoRádió korpusz hivatalos stílusú, szerkesztett szövegeket tartalmaz. A mérések eredményéhez lásd a 3. táblázatot.

FIN	+1	+2	+3	+4	+5	+6	+7
MNSZ2	7527308	163993	5126	1193	267	101	27
Inforádió	23552	220	-	-	-	-	-
MNSZ2%	97,78%	2,13%	0,0666%	0,015%	0,003%	0,001%	3,5e-4%
Inforádió%	99,999%	0,001%	-	-	-	-	-

3. táblázat. A finit ige és a tőle jobbra elhelyezett igekötőjének távolsága – szerkesztett szövegekben 99,9%-ban közvetlenül az ige után, szerkesztetlen szövegekben 99%-ban maximum két token távolságra helyezkedik el az igekötő

Az eredmények tehát azt mutatják, hogy az igekötő nagyon ritkán kerül 2 tokennél távolabbra jobbra az igéjétől. Két extrém példa az MNSZ2 korpuszból:

(1) Azért **mentem** egy kicsit a pop zene fele **el**, mert szeretem a nívós, könnyed jó popzenét.

(2) 27 gyereket **vitt** egy feltehetően részeg buszsofőr Szentesen még csütörtökön egy sportrendezvény után **vissza** az iskolába.

A posztverbális igekötők pozíció szerinti eloszlásában az a tendencia fedezhető fel, hogy legtávolabb a testes igekötők kerülhetnek, amelyek határozószóként is funkcionálnak (pl. *haza*, *vissza*). Ezek jellemzően nem befolyásolják az ige vonzatkeretét. A legkevésbé eltávolodó igekötők grammatikalizált, rövid, prototipikus igekötők, amelyek eloszlásukat tekintve nagyon hasonló százalékokat produkálnak (lásd a 4. táblázatot)[3].

A magyar helyesírás szerint egy igekötő nem előzheti meg közvetlenül az igét, amelyhez tartozik – ekkor egy szót alkot az igével –, ezért ezt a pozíció nem része a kiértékelésnek. Az igét közvetlenül megelőző pozícióban szereplő igekötő vagy egy másik igei elem (pl. az ige infinitívuszi vonzatának) igekötője, vagy elírás eredménye (ekkor valóban az igéé).

[3] A preverbális igekötők pozíció szerinti eloszlását lásd [8]

	-2	0	+1	+2	+3
meg, ki, be, le, fel, föl, el, át, rá	0,49%	58,5%	40%	1%	0,01%

4. táblázat. Néhány gyakori igekötő távolsága a finit igétől – az igekötők 98,5%-a az igén vagy közvetlenül utána áll, csak 1,01% távolodik el jobban (MNSZ2)

2.3. Az infinitívusz és a tőle jobbra elhelyezkedő igekötője

Az infinitívusszal kapcsolatos méréseinket a Pázmány Korpuszon [7] végeztük. Mivel ez web-alapú korpusz, még az MNSZ2-nél is nagyobb arányban tartalmaz szerkesztetlen szövegeket (a kommentkorpusz mérete 2 millió token). Emiatt várható, hogy az infinit ige és a hozzá tartozó igekötő gyakran több szónyi távolságban álljanak egymástól. Az eredményeink mégis azt mutatják, hogy igekötő az esetek 86%-ában közvetlenül az infinit igealak után áll (lásd az 5. táblázatot).

INF [...] IK	db.	%
össz.	717	
+1	619	86,3
+2	52	7,3
+3	35	4,9
>+3	11	1,5

5. táblázat. Az infinitívusz és a tőle jobbra elhelyezkedő igekötőjének távolsága – 93,6%-ban maximum két token van köztük

FIN [...] INF	db.	%
össz.	727562	
+1	652778	89,7
+2	47669	6,6
>+2	27115	3,7

6. táblázat. A finit ige és a tőle jobbra elhelyezkedő infinitívuszi vonzatának távolsága – 96,3%-ban maximum két token van köztük

A kiugróan gyakori +1 pozícióban még sok prototipikus igekötőt találunk, pl. *iparkodott ellentétet* ***mutatni ki***, *javasolt a lapokat lazán* ***helyezni el***. A +2 pozícióról elmondható, hogy az infinitívusz és annak igekötője között csak finit ige állhat, és – bár van példa prototipikus igekötőre, pl. *épp* ***foglalni*** *akartam* ***le*** *a buszt* – nagyobb arányban jelennek meg a testes igekötők (pl. *már* ***indulni*** *akartam* ***vissza***). A nagyon ritka +3 pozícióban testes igekötők állnak, pl. *de már* ***jönni*** *kellett sajnos* ***haza***. A +4 és +5 pozícióra mindössze 15 példát találtunk, ez statisztikailag irreleváns mennyiség. Az itt álló igekötők nem befolyásolják az ige vonzatkeretét (csak az ige által kifejezett mozgás irányát módosítják), pl. ***vinni*** *kell a kamerát* ***el***, ***menekülni*** *akartak a városon keresztül* ***vissza***.

2.4. A finit ige és a tőle jobbra elhelyezkedő infinitívuszi vonzata

Kimértük azt is, hogy az infinitívuszi vonzat mint vonzatkeret-egyértelműsítő elem milyen távol állhat a főigétől. A 6. táblázatban látható, hogy az esetek 89%-ában az infinitívusz közvetlenül a finit ige után áll, 6,5%-ban egy szót enged

maga elé. Az eredmény alapján a legtöbb esetben a főige elemzési ablakába esik az infinitívuszi vonzat, ezzel elősegítve a vonzatkeret-egyértelműsítést.

A fenti eredményeket összefoglalva elmondható, hogy ha az igekötő a finit igétől jobbra helyezkedik el, akkor a legtöbb esetben beleesik az ige 1.1. fejezetben említett elemzési ablakába, tehát az ige elemzésének pillanatában elérhető az elemző számára a vonzatkeret-egyértelműsítéshez. Ugyanerre az eredményre jutottunk az infinitívusztól jobbra elhelyezdedő igekötő és a finit igétől jobbra elhelyezkedő infinitívuszi vonzat esetében is.

3. VFrame

Eredményeink alátámasztják, hogy egy viszonylag kisméretű előretekintő elemzési ablak elegendő ahhoz, hogy a vonzatkeret-egyértelműsítés az igei elem elemzésekor megtörténhessen, hiszen – amint a 2. fejezet eredményei mutatták – az igekötő és az infinitívuszi vonzat az esetek legnagyobb részében elérhető az igei elem számára (a tározóban vagy az ablakban). A következőkben a méréseink alapján létrehozott *VFrame* keresőeljárást ismertetjük, amely az igekötők igei elemekhez kapcsolásával segít előhívni a mondatban előforduló finit és infinit igék megfelelő vonzatkeretét.

Az igei elemet megelőző összes, és az azt követő néhány token ismerete egyértelműsíti a vonzatkeretet az igekötő és az infinitívusz vonzat tekintetében, mely szükséges, de nem elégséges. A 2.1. fejezetben ismertetett öt igeosztály egységes kezeléséről a VFrame gondoskodik, amely minden igei elem összes igekötő–infinitívuszi vonzat kombinációját kezeli (azokat az eseteket is, amikor nincs igekötő és/vagy infinitívuszi vonzat). A VFrame szerkezetét az 1. ábra mutatja.

$$\textsc{VFrame}\begin{bmatrix}\textsc{Irány} = > \mid < \\ \textsc{Igekötő} = \textit{lehetséges igekötők halmaza} \mid X \mid \textit{talált token} \\ \textsc{Infinitívusz} = ? \mid X \mid \textit{talált} \\ \textsc{Találati függvény} = \textit{találatkor vagy a sikertelen keresés végén fut le} \\ \textsc{Egyéb}\begin{bmatrix}\textsc{Tő} = \textit{az ige töve} \\ \textsc{megszorítási függvény} = \textit{a találatok megszorítási szabályai}\end{bmatrix}\end{bmatrix}$$

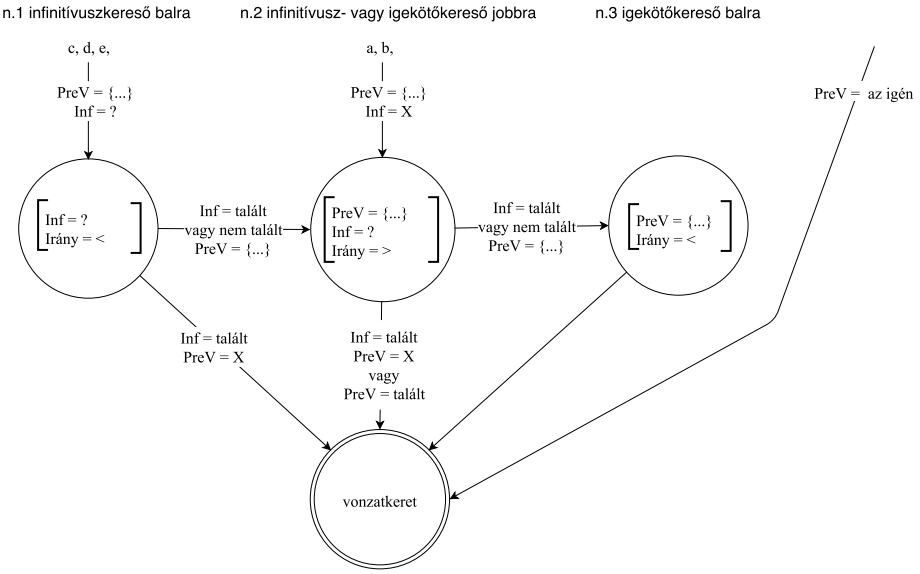
1. ábra. A *VFrame* keresőeljárás architektúrája

A VFrame **Irány** jegye a keresés aktuális irányát mutatja – balra a tározóban vagy jobbra az ablakban. Az **Igekötő** jegy az igei elemmel kompatibilis *összes lehetséges igekötő halmazát* tartalmazza (tekintet nélkül arra, hogy az adott igekötő kizárja-e az infinitívuszi vonzat meglétét), vagy X-et (ha az adott igei elemnek semmilyen igekötője nem lehet), vagy az igei elem az aktuális mondatban már *megtalált igekötőjét*[4]. Az **Infinitívusz** jegy jelzi, hogy az igei elemnek lehet-e infinitívuszi vonzata ($?$) vagy sem (X). Ha az elemző talál infinitívuszi vonzatot,

[4] Abban az esetben, ha az igekötő az igén van, az elemző kihagyja a VFrame keresőeljárást, és elindítja a vonzatok keresését.

akkor azt az Infinitívusz jegy *Talált* állása jelzi. A **Találati függvény** tartalmazza azt a függvényt, amelyet az elemző egy – a VFrame szempontjából fontos – elem megtalálásakor, vagy annak hiányában meghív. A Találati függvény kezeli továbbá a különböző keresőeljárások állapotai közötti átmeneteket is (lást a 2. ábrát). A **Megszorítási függvény** kezeli a találat megszorításait (pl. ha az INF és a PreV kölcsönösen kizárják egymást).

A VFrame aktuális tartalma alapján képes arra, hogy a megfelelő keresőeljárások elindításával egyértelműsítse az aktuális vonzatkeretet. A mondatban több olyan elem is lehet, amelynek lehet igekötője, sőt lehetséges igekötőik halmazában is lehet átfedés, a VFrame keresőeljárással az igék, igekötők és infinitívuszi vonzatok helyesen kapcsolhatók össze. A keresőeljárások sorozata egy véges állapotú automata segítségével írható le három valódi állapottal. A 2. ábra mutatja a folyamat lépéseit[5].



2. ábra. A *VFrame* keresőeljárás állapotainak véges állapotú automata reprezentációja, amely lefedi a 2.1. fejezetben ismertetett öt igeosztályt

4. Problémás esetek

Az igekötőkkel és infinitívuszokkal kapcsolatban egyéb problémák is felmerülhetnek, amelyek megnehezíthetik az igekötő–igei elem vagy infinitívuszi vonzat-igei elem összekapcsolását. Ebben a fejezetben a problémás esetek lajstromba vételével foglalkozunk. A VFrame keresőeljárás önmagában csak az elsőre – a több infinitívuszt tartalmazó mondatok problémájára – nyújt megoldást.

[5] A VFrame pontos működéséről és implementációjáról lásd [10].

4.1. Több infinitívusz

Az olyan példákban, ahol egynél több infinitívusz jelenik meg az igei komplexumban, az infinitívuszok jellemzően egymás mellett állnak, pl. *el kell kezdeni keringőzni tanulni*, *el fogod tudni dönteni*. Azonban arra is vannak példák, hogy az egyik infinitívusz az igei komplexum élére kerül, pl. *pisilni el tudtál menni*.

A VFrame keresőeljárás a több infinitívuszt tartalmazó mondatokkal is megbirkózik. Minden igei elem, így a főige és a mondatban szereplő infinitívuszi vonzatok egyaránt elindítják a saját VFrame keresőeljárásukat a megfelelő beállításokkal. Az olyan mondatok esetén, amelyeket a vonzatok nem természetes sorrendje miatt az ember is nehezen elemez, az elemző visszalépéssel és újraelemzéssel alakítja ki a megfelelő igei elem–igekötő és igei elem–infinitívuszi vonzat viszonyokat.

4.2. A homonímia

Két gyakori igekötő, a *meg* és a *ki* esetében gondot jelent az, hogy mindkét szó homonim, és sokszor hibás annotációval szerepel a korpuszban. A *meg* gyakran IK (azaz igekötő) címkét kap akkor is, ha mellérendelő kötőszó, a *ki* igekötő pedig gyakran keveredik az azonos alakú vonatkozó- illetve kérdő névmással. A hibásan annotált szavak automatikus azonosítását nehezítik az olyan esetek, amikor ezek valóban létező kombinációt alkothatnának az igével, ráadásul olyan pozícióban is állnak, amely az igekötők számára is elérhető (lásd az 3. példát).

(3) a. *akkor csak lámpát kell vennem* ***meg*** *rácsot*
a *meg* igekötőként a létező *meg+vesz* igét eredményezheti

b. *az mennyibe fog kerülni és* ***ki*** *fogja rá adni a pénzt*
a *ki* igekötőként a *ki+ad* létező igét eredményezheti

4.3. „Megírni meg kell”

A korpuszból kinyert mondatokban több mint 200 példát találtunk egy különleges szerkezetre, amelyben látszólag nem tartozik ige az igekötőhöz. A szerkezet egy infinitívuszból, egy finit igéből (jellemzően segédigéből) és egy olyan igekötőből áll, amely az infinitívuszon is megjelenő igekötő hangsúlyos alakja. Például: ***elképzelni*** *bármit* ***el*** *lehet,* ***becsajozni be*** *tudnék.*

4.4. Más igei elemek

A többi, vonzatkerettel és igekötővel rendelkező igei elem (melléknévi és határozói igenév) is rendelkezik VFrame keresőeljárással, ám ezek esetében számos más probléma is felmerül – például a befejezett melléknévi igenév–melléknév–múlt idejű ige szófaji többértelműség kezelése –, amelyek további kutatások tárgyát képezik. Ezen igei elemek esetében a VFrame kiegészül egy olyan megszorítással, amely szerint az igekötőt vagy az infinitívuszi elemet az igenevet tartalmazó NP határain belül és csak balra (a tározóban) keresi.

5. Összefoglalás

Korpuszméréseink alapján bizonyítottuk, hogy az AnaGramma elemzőrendszer keretein belül a finit ige–igekötő kapcsolat létrehozása mellett [10] az infinitívusz–igekötő és a finit ige–infinitívuszi vonzat kapcsolatok létrehozásához is elegendő a feltételezett két token méretű elemzési ablak használata. A tározó és az ablak segítségével a VFrame keresőeljárás a mondatban szereplő igei elemeket (finit és infinit igéket) valamint az igekötőket a megfelelő módon kapcsolja össze.

Az aktuális finit ige–igekötő–infinitívuszi vonzat kapcsolatok létrejötte után elindulnak a megfelelő vonzatkeresők, amelyek mind a tározóban, mind a mondat hátralévő részében keresik a vonzatkeret elemeit. Amennyiben a VFrame nem egyértelműsíti teljesen a vonzatkeretet (mert egy ige ugyanazzal a finit ige–igekötő–infinitívuszi vonzat viszonnyal többféle vonzatkerettel is rendelkezhet), akkor az összes ennek megfelelő vonzatkeret vonzatkeresője elindul. Ekkor a mondatban aktuálisan szereplő többi vonzat egyértelműsíti a vonzatkeretet.

Hivatkozások

1. Prószéky, G., Indig, B., Vadász, N.: Performanciaalapú elemző magyar szövegek számítógépes megértéséhez. In Bence, K., ed.: "Szavad ne feledd!": Tanulmányok Bánréti Zoltán tiszteletére. MTA NYTI, Budapest (2016) 223–232
2. Indig, B., Vadász, N., Kalivoda, Á.: Decreasing Entropy: How Wide to Open the Window? In Martín-Vide, C., Mizuki, T., Vega-Rodríguez, M.A., eds.: Theory and Practice of Natural Computing: 5th International Conference, TPNC 2016, Sendai, Japan, December 12-13, 2016, Proceedings, Cham, Springer (2016) 137–148
3. Prószéky, G., Indig, B.: Magyar szövegek pszicholingvisztikai indíttatású elemzése számítógéppel. Alkalmazott nyelvtudomány **15**(1-2) (2015) 29–44
4. Turi, Z., Németh, D., Hoffmann, I.: Nyelv és emlékezet. In Pléh, C., Lukács, A., eds.: Pszicholingvisztika 2. Akadémiai Kiadó, Budapest (2014) 743–776
5. Frazier, L., Fodor, J.D.: The Sausage Machine: A New Two-Stage Parsing Model. Cognition **6**(4) (1978) 291–325
6. Oravecz, C., Váradi, T., Sass, B.: The Hungarian Gigaword Corpus. In Calzolari, N., et al., eds.: Proceedings of the 9th International Conference on Language Resources and Evaluation, May 26-31, 2014, Reykjavik, Iceland, ELRA 1719–1723
7. Endrédy, I.: Nyelvtechnológiai algoritmusok korpuszok automatikus építéséhez és pontosabb feldolgozásukhoz (2016) PhD disszertáció. PPKE-ITK.
8. Kalivoda, Á.: A magyar igei komplexumok vizsgálata (2016) MA szakdolgozat. PPKE-BTK. `https://github.com/kagnes/hungarian_verbal_complex`.
9. É. Kiss, K.: Az ige utáni szabad szórend magyarázata. Nyelvtudományi Közlemények **104** (2007) 124–152
10. Indig, B., Vadász, N.: Windows in Human Parsing – How Far can a Preverb Go? In Tadić, M., Bekavac, B., eds.: Tenth International Conference on Natural Language Processing (HrTAL2016) 2016, Dubrovnik, Croatia, September 29-30, 2016, Proceedings, Cham, Springer (2016) (Elfogadva, nyomtatás alatt)

Főnévi események automatikus detektálása függőségi elemző és WordNet alkalmazásával magyar nyelvű szövegeken

Subecz Zoltán[1]

[1] Pallasz Athéné Egyetem, Kecskemét
subecz@szolf.hu

Kivonat. A természetes szövegekből történő információkinyerés egyik fontos részterülete a névelemek azonosítása mellett az események detektálása. Szövegekben lévő események detektálása és analizálása fontos szerepet tölt be számos számítógépes nyelvészeti alkalmazásban, mint például a kivonatolás és a válaszkeresés. A szövegekben a legtöbb esemény igékhez kapcsolódik, és az igék általában eseményeket jelölnek. De az igéken kívül lehetnek események más szófajú szavak is pl. főnevek, igenevek stb. Munkánkban a szövegekben megtalálható főnévi események detektálásával foglalkoztunk. Jelen tanulmányunkban bemutatjuk gazdag jellemzőtéren alapuló gépi tanuló megközelítésünket, amely automatikusan képes magyar nyelvű szövegekben főnévi események detektálására függőségi elemző és WordNet alkalmazásával. A jellemzők mellé kiegészítő módszereket is alkalmaztunk, amelyek javították az eredményeket és a futási időt. Algoritmusainkat tesztadatbázisokon kiértékelve versenyképes eredményeket érnek el az eddig bemutatott angol és más nyelvű eredményekkel összehasonlítva.

Kulcsszavak: információkinyerés, eseménydetektálás, főnévi események detektálása, WordNet, függőségi elemzés

1 Bevezetés

A természetes szövegekből történő információkinyerés egyik fontos részterülete a névelemek azonosítása mellett az események detektálása [7]. Szövegekben lévő események detektálása és analizálása fontos szerepet tölt be számos számítógépes nyelvészeti alkalmazásban, mint például a kivonatolás és a válaszkeresés. A szövegekben lévő események felismerése, analizálása, és hogy hogyan viszonyulnak egymáshoz időben, fontos a szöveg tartalmának megismerésében.

Az esemény, ami történik egy adott helyen és időben. A szövegekben a legtöbb esemény igékhez kapcsolódik, és az igék általában eseményeket jelölnek. De az igéken kívül lehetnek események más szófajú szavak is pl. főnevek, igenevek stb. Munkánkban a szövegekben megtalálható főnévi események detektálásával foglalkoztunk. Vannak olyan szavak (pl. írás), amelyek egyes mondatokban események, másokban pedig nem, ezért a szavak szövegkörnyezetét is elemezni kell. Jelen tanulmányunkban

bemutatjuk gazdag jellemzőtéren alapuló gépi tanuló megközelítésünket, amely automatikusan képes magyar nyelvű szövegekben főnévi események detektálására függőségi elemző és WordNet alkalmazásával. A rendszer bemenete egy token-szinten címkézett tanító korpusz. Modellünk jelöltjei a mondatok főnevei voltak.

A feladatokhoz gazdag jellemzőkészleten alapuló osztályozót használtunk. A jellemzők mellé kiegészítő módszereket is alkalmaztunk, amelyek javították az eredményeket és a futási időt. Módszerünket a Szeged Korpusz öt különböző doménjén vizsgáltuk meg.

Angol nyelvű szövegekre általában *konstituensfa alapú* szintaktikai elemzőt használnak az elő-feldolgozásnál az angol nyelv erősen konfiguratív tulajdonsága miatt, ahol is a legtöbb mondatszintű szintaktikai információt a szórenddel fejeznek ki. Ezzel ellentétben a magyar nyelv gazdag morfológiával és szabad szórenddel rendelkezik. A *függőségi fákkal* dolgozó elemzők különösen jól használhatóak szabad szórendű nyelvek elemzésére, így a magyarra is. Ezek ugyanis könnyebben teszik lehetővé az egymással nem szomszédos, de összetartozó szavak összekapcsolását is. Ezért mi a magyar nyelvű szövegeinkre *függőségi fákkal dolgozó elemzőt* használtunk.

Megoldásunkban a vizsgált szavak szemantikai jellemzéséhez felhasználtuk a magyar *WordNet*-et [10]. Mivel egy szóalakhoz több jelentés is tartozhat a WordNet-ben, ezért az egyes jelentések között egyértelműsítést végeztünk a *Lesk algoritmussal* [8].

Algoritmusainkat tesztadatbázisokon kiértékelve, versenyképes eredményeket érnek el az eddig bemutatott angol és más nyelvű eredményekkel összehasonlítva.

2 Kapcsolódó munkák

Az EVITA [13] volt az első esemény felismerő eszközök egyike. Az eseményeket nyelvészeti és statisztikai technikák kombinálásának segítségével ismeri fel. Nyelvészeti ismereteken alapuló szabályokat használ fő jellemzőként. A főnévi esemény felismeréshez WordNet osztályokat is használ, valamint a főnevek szemantikai egyértelműsítésére Bayes osztályozót alkalmaz.

Boguraev és társa [2] gépi tanuláson alapuló módszert mutattak be automatikus esemény-annotáláshoz. A feladatot osztályozásra visszavezetve, RRM (robust risk minimization) osztályozót alkalmaztak. Jellemzőkként lexikai, morfológiai és szintaktikai chunk típusokat használtak két- és háromelemű ablakokban vizsgálva.

Bethard és társa [1] esemény felismerésre fejlesztették a STEP rendszert. Szintaktikai és szemantikai jellemzőket alkalmaztak és az esemény felismerési feladatot osztályozásként oldották meg. Gazdag jellemző készletet építettek be: lexikai, morfológiai, szintaktikai függőségi és választott WordNet osztályokat. E jellemzőkre alapozva Support Vector Machine (SVM) modellt implementáltak.

Llorens és társa [9] bemutattak egy kiértékelést esemény felismerésre. Szemantikai szerepeket adtak meg jellemzőként és CRF (Conditional Random Field) modellt építettek események felismeréséhez.

Jeong és társa [6] függőségi elemzőt használtak, de csak a jelölt főnév és a közvetlenül ahhoz kapcsolódó ige közötti kapcsolatot vizsgálták. Kombinált jellemzőket építettek be, az ige és a hozzá tartozó kapcsolat-típus párokat alkalmazva. A

WordNetet is használták, de jelentésegyértelműsítés nélkül. A MaxEnt osztályozási algoritmust a következő jellemzőkészlettel implementálták: felszíni, lexikai, szemantikai, függőségi alapú jellemzők. A jellemzőket a Kullback-Leibler divergencia módszerrel súlyozták.

Olasz szövegekre Caselli és társa [3] csak igéből képzett főnévi eseményekkel foglalkoztak, amihez a Weka döntési fa alapú osztályozót használták. Vizsgálták a jelölt argumentum struktúráját, az aspektuális módosítókat, a jelölt előtti és utáni 3-3 szó szófaját.

Spanyol szövegekre Peris és társa [11] szintén csak igéből képzett eseményekkel foglalkoztak. Osztályozásra a Weka döntési fa osztályozóját alkalmazták és külső főnévi lexikont használtak fel. Függőségi elemzőt alkalmaztak, de csak a jelölt főnév és a közvetlenül ahhoz kapcsolódó ige közötti kapcsolatot vizsgálták. Felhasználták a jelölt argumentum struktúráját is.

Német nyelvű szövegekre Gorzitze és társa [5] bootstrapping módszert alkalmaztak események felismerésre. Idővel kapcsolatos kifejezéseket és aspektuális igéket kerestek a jelölt közelében. A jelölt és a közvetlenül ahhoz kapcsolódó ige kapcsolatát vizsgálták és szabály alapú függőségi elemzőt használtak.

Magyar szövegekre Subecz [14] detektált eseményeket, de csak igei és főnévi igenévi eseményekkel foglalkozott. A következő jellemző készletet használta: felszíni, lexikai, morfológiai, szintaktikai, WordNet és frekvencia jellemzők. Ezen jellemzők mellett szabály alapú módszereket is alkalmazott.

3 Események, főnévi események

A szövegekben a legtöbb esemény igékhez kapcsolódik, és az igék általában eseményeket jelölnek. De az igéken kívül lehetnek események más szófajú szavak is például főnevek, igenevek. Munkánkban a szövegekben megtalálható főnévi események detektálásával foglalkoztunk. Példák főnévi eseményekre: futás, építés, írás, háború, ünnepség.

A főnévi eseményeknek két nagy csoportja van: igéből képzettek (deverbális) és nem igéből képzettek (nem deverbális). Példa igéből képzett eseményekre: futás, írás. Példa nem igékből képzett eseményre: háború. Az igéből képzett főnevek két fő fajtája az események és az eredmények, amelyeknél gyakori a kétértelműség is. Vannak olyan szavak (például írás), amelyek egyes mondatokban események, másokban pedig eredmények.

Például az *írás* főnév a következő mondatban esemény: *Az írás 5 órakor kezdődött.* Viszont a következő mondatban nem esemény, hanem eredmény: *Elolvastuk az írást.* A többértelműség miatt nem elég a szóalak vizsgálata, a szövegkörnyezetet is elemezni kell.

4 Környezet

Alkalmazásunkban a Szeged Korpusz [4] egy részét használtuk fel a következő területekről: *üzleti rövidhírek, szépirodalom-fogalmazás, számítógépes szövegek, újsághí-*

rek, jogi szövegek. Tanításhoz és kiértékeléshez tízszeres keresztvalidációt alkalmaztunk. A mondatokat két nyelvész annotálta, az annotátorok közötti egyetértés Kappa = 0,7 volt.

A feladatokat *bináris osztályozásra* vezettük vissza. Az osztályozáshoz a *Weka* programcsomagnak[1] a J48-as döntési fa elemzőjét használtuk fel. A feladathoz alkalmaztuk még a Magyarlanc 2.0 programcsomagot is [16]. A csomagot magyar szövegek mondatokra és szavakra bontására, a szavak morfológiai elemzésére, majd szófaji egyértelműsítésére, és mondatok függőségi nyelvtan szerinti szintaktikai elemzésére alkalmaztuk. A Magyarlanc programcsomag is készít a szavakhoz morfológiai elemzést, de a HunMorph[15] elemzőcsomag sok esetben részletesebb elemzést ad, ezért ezt is felhasználtuk. Így a feladathoz két morfológiai elemzőt is alkalmaztunk.

Ahogy láttuk a kapcsolódó munkáknál, mások is használtak függőségi elemzést. De az elemzőfában mindenki csak a jelölt és a vele közvetlen kapcsolatban lévő szavakat vizsgálta. Mi vizsgáltuk a jelölt és a fában tőle távolabbi igék kapcsolatát is. Modellünk jelöltjei a mondatok főnevei voltak. Számos esetben a jelölt alá egy részfa tartozott a függőségi fában.

A vizsgált főnevek szemantikai jellemzéséhez a magyar *WordNet*-et[10] alkalmaztuk. A WordNet hiperním hierarchiájában található szemantikai kapcsolatokat használtuk fel.

A szavak közötti szintaktikai kapcsolatok alapján a mondatok egy *függőségi fát* alkotnak. A fa legfelső eleme a *Root*. A fa *csomópontjaiban* vannak a mondat szavai, az *ágak* a szavak közötti *szintaktikai kapcsolatokat* reprezentálják. Ha a jelölt több szóból áll, akkor ezek a szavak egy részfát alkotnak a mondat fáján belül. A részfa a kiemelt szaván (fejszó, headword) keresztül kapcsolódik a fa többi részéhez.

Statisztikai adatok: A vizsgált korpusz 10000 mondatot tartalmaz. Jelöltek száma (főnevek): 48388 db. Pozitív jelöltek száma (esemény főnevek): 7626 db. A jelölteket két fő részre osztottuk a hasonló tulajdonságok alapján. Az egyik csoportba az igéből képzett főnevek, a másikba a többi főnév került. Igéből képzett jelöltek: 5325 db. Igéből képzett pozitív jelöltek: 4169 db. Nem igéből képzett jelöltek: 43063 db. Nem igéből képzett pozitív jelöltek: 3457 db.

5 Az osztályozás bemutatása

Az osztályozáshoz bináris osztályozót használtunk. A mondatok főnevei voltak a jelöltek. Ezek az elemzőfában egy-egy csomópontot jelentenek.

5.1 Jellemzőkészlet

A tanító és a kiértékelő halmazon a jelöltekhez jellemzőket vettünk fel. Módszerünket gazdag jellemzőtérrel valósítottuk meg. Az eseménydetektálással kapcsolatos feladatokban gyakran használt jellemzőket mi is alkalmaztuk. Ezeken kívül újakkal is kibővítettük a jellemzőkészletünket. Az új jellemzőket a magyar szövegek tulajdonságai

[1] http://www.cs.waikato.ac.nz/ml/weka/

alapján választottuk ki. A jellemzőkhöz felhasználtuk a függőségi elemzőfát és a magyar WordNet-et is. A fő jellemző csoportokat több részre bontottunk, mivel a részek hatását külön-külön vizsgáljuk majd. Ezen csoportok közül a következők voltak az új, máshol ezen a területen nem látott jellemzőcsoportok: a *két Morfológiai elemző együtt*, az *elemzőfa 1-2*, *szózsák 1-3, WordNet jellemzők-2-4.*

Felszíni jellemzők: *Bigramok, trigramok*: A vizsgált szavak végén lévő 2-es, 3-as betűcsoportok. *PositionInSentence*: a jelölt hányadik szó a mondatban. *NagyBetuNemMondatElejen*: Azok a nagybetűs szavak, amelyek nem a mondat elején állnak legtöbbször névelemek. Így ez a jellemző utalhat a nem-esemény jellegre.

Morfológiai jellemzők-1: Mivel a magyar nyelv igen gazdag morfológiával rendelkezik, ezért számos morfológia-alapú jellemzőt definiáltunk. Ebben a csoportban a Magyarlanc morfológiai elemzőjét használtuk fel. Jellemzőként definiáltuk az eseményjelöltek MSD morfológiai kódját, felhasználva a következő morfológiai jegyeket: *típus*(SubPos), *mód*(Mood), *eset*(Cas), *idő*(Tense), *személy*(PerP), *szám*(Num), *hatá-rozottság(*Def). További jellemzők: *Lemma*: a jelölt lemmája. *hasVerbRoot*: igéből képzett-e a jelölt. *SzofajElotte* és *SzofajUtana*: a jelölt előtti és utáni szó szófaja. *LegkozelebbiIgeMondatbanLemma*: a jelölthöz a mondatban legközelebb álló ige lemmája. *Igeto*: igéből képzett főnév esetén az alapige.

Morfológiai jellemzők-2: Ebben a csoportban a HunMorph morfológiai elemzőjét használtuk fel. *IgetoVan*: van-e igető. *IgebolFonevKepzo*: Igéből képzett főneveknél a képző. *IgeToHunMorph*: igéből képzett főnév esetén az alapige.

Morfológiai jellemzők-3: A Magyarlanc és a HunMorph morfológiai elemző is a többjelentésű szavak esetén minden jelentéshez megadnak külön morfológiai elemzést. Ebben a csoportban mindkét elemző esetén, egyértelműsítés nélkül, megadtuk a jelöltekhez minden jelentéshez tartozó ragokat, képzőket, jeleket.

Elemzőfa jellemzők-1: Ezeket a jellemzőket az függőségi elemzőfa alapján készítettük. *JeloltEdgeType*: A jelölt és az elemzőfában a felette levő szó közötti kapcsolat típusa. *JeloltEdgeTypeNE*: A jelölt NE (névelem) típussal kapcsolódik-e a felette levő szóhoz. Ez utal a jelölt nem esemény jellegére. *JeloltFelettLemmaFaban*: A jelölt feletti szó lemmája az elemzőfában. *JeloltFelettIgeLemmaFaban*: Az elemzőfában közvetlenül a jelölt felett lévő ige (ha van) lemmája. *KozvetlenSzintaktikaiKapcsolat*: Ha a jelölt fölött van közvetlenül ige az elemzőfában, akkor a kettő közötti szintaktikai kapcsolat típusa. *LegkozelebbiIgeFeletteFabanLemma, LegkozelebbiIgeFelette-TavolsagFaban*: Az elemzőfában jelölt feletti legközelebbi ige lemmája és annak távolsága a fában a jelölttől. *JeloltReszfaTokenekSzama*: Az elemzőfában a jelölt alá tartozó részfa elemeinek száma. *FeletteSzoEdgeType*: Az elemzőfában a jelölt feletti szó és az a feletti szó közötti kapcsolat típusa. A Magyarlanc elemzőnél az időhatározók, időbeliséget kifejező szavak az események felett helyezkednek el, ezért ezek jelenléte utalhat a jelölt eseményjellegére.

Elemzőfa jellemzők-2: Ha a jelölt nem közvetlenül kapcsolódik a felette levő igéhez az elemzőfában, akkor részletesen jellemeztük a jelölt és az ige közötti útvonalat. *SzófajÚtvonal*: Egymás után írtuk a jelölt és az ige közötti csomópontok szófaját. Például: C↑S↑V↑C↑V↑V. *Lemmaútvonal*: Hasonlóan az előzőhöz a jelölt és az ige közötti lemmákat írtuk egymás után. Például: napoztatás↑és↑törölgetés↑hajszárító↑megszárít. *SzintaktikaiKapcsolat-Útvonal*: A jelölt és az

ige közötti útvonalon a szintaktikai kapcsolatok típusai egymás után. Például: OBL↑COORD↑SUBJ↑COORD↑CONJ↑.

Szózsák jellemzők-1: Szózsák modellt használtuk fel szócsoportok jellemzésére. *ReszfaLemmakSzozsakAtlag:* A jelölt alatt lévő részfa szavainak lemmáit reprezentáltuk szózsák modellel. A tanító halmazon minden lemmához kiszámítottuk, hogy milyen arányban tartozott pozitív jelölt a részfájához. Majd minden jelölthöz kiszámítottuk a részfáját alkotó lemmákhoz tartozó arányok átlagát. Nagy átlag arra utal, hogy a jelölt részfájában fontos szavak vannak az eseményjelleg szempontjából. *ReszfaLemmakSzozsakLegnagyobb*: Hasonló az előzőhöz, de itt minden jelöltnél a részfájához tartozó lemmák közül azt választottuk ki, amelyikhez legnagyobb valószínűség tartozott. Nagy maximális valószínűség utal arra, hogy a jelölt részfájában van legalább egy olyan lemma, ami erősen fontos az eseményjelleg szempontjából. Ez a jellemző segít a részfa egy-egy fontos szavának felismerésében. *KozvetlenAlattaLemmakSzozsakAtlag* és *KozvetlenAlattaLemmakSzozsakLegnagyobb*: Az előzőkhöz hasonló, de itt nem a jelölt részfájához tartozó minden szót vizsgáltuk, hanem csak a részfa azon szavait, amelyek szintaktikailag kapcsolódnak a jelölthöz az elemzőfában. *KozvetlenAlattaEdgeTypeSzozsakAtlag* és *KozvetlenAlattaEdgeTypeSzozsakLegnagyobb*: Az előzőhöz hasonlóan, itt a jelölt és a hozzá szintaktikailag kapcsolódó szavak közötti kapcsolat típusát vizsgáltuk. *LemmaParseTreePathIgeigLemmakSzozsakAtlag* és *LemmaParseTreePathIgeigLemmakSzozsakLegnagyobb*: Ezeknél a szózsákba a jelölt és az elemzőfában felette lévő ige közötti útvonalon található lemmák kerültek.

Szózsák jellemzők-2: Ezekhez a jellemzőkhöz a lemmák a mondatból és nem az elemzőfából lettek kigyűjtve a szózsákba. *MondatbanKornyezet-N-LemmakSzozsakAtlag* és *MondatbanKornyezet-N-LemmakSzozsakLegnagyobb*: A mondatban a jelölt N távolságú környezetét jellemeztük szózsák modellel, N=3 és N=5 esetekben.

WordNet jellemzők: Ezekhez a jellemzőkhöz felhasználtuk a magyar WordNet [10] hiperním hierarchiájában található szemantikai kapcsolatokat. Mivel egy szóalakhoz több jelentés is tartozhat a WordNet-ben, ezért az egyes jelentések között egyértelműsítést végeztünk a Lesk algoritmussal[8] A WordNetben a synsetekhez definíció és példamondatok tartoznak. Az algoritmus alapján, többjelentésű eseményjelölt esetén megszámoltuk, hogy az eseményjelölt szintaktikai környezetében lévő szavak közül hány található meg az egyes WordNet jelentések definíciói és példamondatai között. Azt a jelentést választottuk, amelyikkel a legtöbb volt közös szó.

WordNet jellemzők-1: *EsemenyszerusegekAlatt*: A magyar WordNetben van egy mesterséges synset, ami alá jellemzően események tartoznak. De vannak események ezen kívül is, és vannak ebben a csoportban olyanok is, amelyek nem igazi események. Ebben a jellemzőben megadtuk, hogy a jelölt beletartozik-e ezen synset hiponím hierarchiájába.

WordNet jellemzők-2: *WordNetSzozsakAtlag* és *WordNetSzozsakLegnagyobb*: A szózsák jellemzőkhöz hasonlóan itt a szózsákba a WordNetben a jelölt hiperním hierarchiájába tartozó szavakat vettük fel. *WordNetSzozsakLegnagyobbSynset*: Megadtuk a jelölt hiperním hierarchiájában lévő synset-ek közül azt, amelyik a legnagyobb arányban tartozik események hiperním hierarchiájába.

WordNet jellemzők-3: *WordNetHipernimSynsetekTanulobol* (bináris): Készítettünk egy halmazt, amibe kigyűjtöttük a tanító halmazból az esemény jelöltek hiperním hierarchiájának synset-jeit. Majd minden jelölthöz megadtuk, hogy a hiperním hierarchiájának synset-jei közül tartozik-e valamelyik ebbe a halmazba.

WordNet jellemzők-4: *WordNetLegjobbLemmakAlatt*: Kigyűjtöttük azokat a lemmákat, amelyek a tanító halmazon legnagyobb arányban voltak események. Majd a jelölteknél jelöltük, hogy a jelölt lemmája alatta van-e valamelyik ilyen kiemelt lemma hiponím hierarchiájának a WordNet-ben.

Szózsák jellemzők-3: Először a *Szózsák jellemzők 1-2* csoportoknál bemutatott minden esethez itt kiválasztottuk a legjobb elemeket a szózsákokból 1-1 halmazba. Azokat, amelyek legnagyobb arányban tartoztak eseményekhez. Majd a következő jellemzőkkel jelöltük, hogy az adott jelölthöz tartozó szózsák tartalmaz-e az adott halmaz elemei közül. *LegjobbWordNetSynsetek*: A jelölt hiperním hierarchiájába tartozó synsetek között van-e ami szerepel a LegjobbWordNetSynsetek halmazban. *LegjobbReszfaLemmak*: A jelölt részfáinak lemmái között van-e olyan lemma, ami szerepel a LegjobbRészfaLemmák halmazban. *LegjobbLemmakUtvonalIgeig*: A jelölt és az elemzőfában a legközelebbi ige közötti lemmák között van-e olyan lemma, ami szerepel a LegjobbÚtvonalLemmák halmazban. *LegjobbMondatbanKornyezet-N-Lemmak*: A mondatban a jelölt N távolságú környezetében van-e olyan lemma, ami szerepel a LegjobbMondatbanKornyezet-N-Lemmak között. Ezt megnéztük N=3 és N=5 esetekre is.

Lista-jellemzők: *FeletteLemmaIdohatarozoListabol*: Listába kigyűjtöttünk gyakori idővel kapcsolatos szavakat. (például: előtt, folyamán) Ezek a szavak alatt az elemzőfában gyakran események vannak. Jellemzőként jelöltük, hogy a jelölt felett van-e ilyen idővel kapcsolatos kifejezés. *FeletteIgeAspektualisListabol*: Listába kigyűjtöttünk gyakori aspektuális igéket (például elkezd, folytatódik). Ezen igék alá tartozó főnevek gyakran események. Jelöltük, hogy a jelölt felett az elemzőfában van-e ilyen ige.

Kombinált jellemzők-2 eleműek: Ezeknél a jellemzőknél az előző jellemzők közül kombináltunk össze kettőt. *JeloltFelettLemmaFaban+JeloltEdge-Type*: Egy szó eseményjellegét gyakran pontosabban jelzi, ha a felette levő lemmát és a kettőjük közötti kapcsolatot együtt vizsgáljuk, mintha csak külön-külön vizsgálnánk azokat. Hasonlóan együtt vizsgáltuk a következőket:
JeloltFelettIgeLemmaFaban+JeloltEdgeTypeOBJ,
JeloltFelettIgeLemmaFaban+JeloltEdgeTypeSUBJ,
JeloltFelettLemmaFaban+LegjobbWordNetSynsetek,
JeloltFelettIgeLemmaFaban+ LegjobbWordNetSynsetek

Kombinált jellemzők - 3 eleműek: Az előző kételemű jellemzőkhöz hasonlóan itt három jellemzőt kombináltunk össze.
JeloltFelettLemmaFaban+EdgeType+WordNetLegjobbSynset,
JeloltFelettIgeLemma-Faban+JeloltEdgeType+WordNetLegjobbSynset,

5.2 További alkalmazott módszerek

A következő módszerek újaknak tekinthetőek, mert ezeken a területeken nem láttuk máshol az alkalmazásukat. Mindegyik hasznos volt az eredménye alapján, így más NLP feladatoknál is hasznosak lehetnek.

Statisztikai arány felhasználása az osztályozásnál. A jelöltekhez a jellemzőket *két módszer* alapján választottuk ki. *Első módszernél* az előző részben bemutatott alapjellemzőket használtuk fel. *Második módszernél* az alapjellemzők helyett a tanító adatokon számított statisztikai arányokat használtuk fel. A tanító halmaz alapján megszámoltuk minden jellemző esethez, hogy hány alkalommal fordult elő és ebből hányszor volt a jelölt *pozitív*. Ezek alapján kiszámítottuk a hozzá tartozó pozitív-arányt. Például ha a *Lemma* jellemzőnél a *Lemma = írás* eset 5-ször fordult elő és ebből 3-szor volt *pozitív eset*, akkor hozzá a 0,6-es pozitív-arány tartozott. Ebben az esetben az osztályozónak a jelöltekhez nem az alapjellemzőt, hanem a hozzá tartozó arányt adtuk meg. Az előző példánál *Lemma-arány* = 0,6. Ezzel *jelentősen csökkentettük az osztályozó vektorterének méretét* az első módszerhez képest és így a *futási időt is*. Ez a kidolgozási időszakban hasznos volt. A két esetet összehasonlítva azt tapasztaltuk, hogy legtöbbször a *valószínűségi módszer* adta a legjobb eredményeket. És a futási idő is *70-80-szor gyorsabb*, mint az alapjellemzők használata esetén.

A jelöltek csoportokra bontása. Az osztályozó hasonló tulajdonságú adathalmazon könnyebben találja meg a szabályokat, mint olyan halmazon, ami sokféle adatot tartalmaz. Ezért érdemes a jelölteket kisebb, hasonló tulajdonságú csoportokra bontani, (ezzel megkönnyíteni az osztályozó döntését). Majd a csoportok eredményeit a TP, TN, FP, FN eredmények alapján összegezni. Ennek megfelelően a jelöltjeinket két fő szempont szerint csoportosítottuk. Első lépésként a jelölteket két csoportra bontottuk: igéből képzett (deverbális) és nem igéből képzett (nem deverbális) főnevek. Hiszen e két csoport tagjai eltérően viselkednek. Az igéből képzett főnevek között sokkal nagyobb arányban vannak események. Másik csoportosítás a jelöltek lemmái alapján történt. Itt 3 alcsoportot képeztünk. Első csoportba azok a lemmák kerültek, amelyek nagy arányban események voltak a tanító halmazon. A másik csoportba a többi jelölt lemmája a tanító halmazról. Harmadik csoportba a kiértékelő halmazon azon jelöltek lemmái, amelyek nem szerepeltek a tanító halmazon. Így összesen 2*3=6 csoportot képeztünk, és mindegyikre külön-külön végeztük el az osztályozást.

Valószínűségi arányok felhasználása az osztályozás eredményeinek javítására. Azokban az esetekben, amikor gyenge eredményt kaptunk, (általában a kis fedés miatt), akkor az osztályozás elvégzése után azon jelölteknél, amelyek a tanító halmazon nagy arányban voltak események, az értékelést a kiértékelő halmazon pozitívra állítottuk.

Majd az eredményeknél látni fogjuk, hogy ezek a kiegészítő módszerek jelentősen javították az eredményeinket és a futási időt.

5.3 Jellemző-esetek számának csökkentése

A *vektortér* méretét *csökkentettük* a következő módszerrel: csak azokat a *jellemző-előfordulásokat* vettük fel az osztályozáshoz, amelyek a tanító halmazon *legalább háromszor* szerepeltek. Ezzel *jelentősen csökkentettük a futási időt* és csak az osztályozás szempontjából jelentéktelen jellemző-előfordulásokat hagytuk ki.

6 Eredmények

A kiértékelés során a pontosság (P), fedés (R) és F-mérték (F) metrikákat használtuk.

6.1 Baseline mérés

Modellünk hatékonyságának vizsgálatához Baseline mérést végeztünk. Ennek keretében a jelöltek közül az igei alapúakat vettük pozitív esetnek a többit pedig negatívnak. Ennek eredménye a következő volt: pontosság: 66,67, fedés: 47,57, F-mérték: 55,52. A további eredményeinken látni fogjuk, hogy *gépi tanulási módszerünk jóval felülteljesítette a Baseline mérés eredményét.*

6.2 Modellünk eredménye

Gépi tanulási módszerünk a következő eredményt érte el a teljes korpuszon az adott jellemzőkészlettel és a kiegészítő módszerekkel: *Pontosság: 79,25, Fedés:67,04, F-mérték: 71,94*. A kiegészítő módszerek alkalmazása nélkül a következő eredményeket kaptuk: Pontosság: 70,32, Fedés: 60,51, F-mérték: 65,03. Látható, hogy a *kiegészítő módszerekkel jelentős javulást tudtunk elérni.* A javulás 80%-át a jelöltek csoportosítása adta, a kisebb részt az osztályozás utáni javításból származott. Ha csak az első szempont szerint csoportosítottunk, akkor azt kaptuk, hogy az igéből képzett főnevek esetén a modell sokkal jobb eredményt ért el (F-mérték: 84,62), mint a nem igékből képzett főneveknél (F-mérték: 39,52).

Modellünket megvizsgáltuk az öt részkorpuszon is. Ezekre az 1. táblázatban látható F-mértékeket kaptuk.

Table 1. Eredmények a részkorpuszokon (%)

Részkorpusz	**F-mérték**
Szépirodalom-fogalmazás	**75,24**
Újsághírek	**76,31**
Üzleti rövidhírek	**75,12**
Számítógépes szövegek	**71,57**
Jogi szövegek	**68,74**

Legjobb eredményünket az *újsághírek* doménen, a legrosszabbat pedig a *jogi szövegeken* kaptuk.

6.3 Eredmények porlasztásos méréssel

Megvizsgáltuk, hogy az egyes **jellemzőcsoportok** hogyan befolyásolják a gépi tanuló-rendszer eredményeit. Ehhez *porlasztásos mérést* végeztünk. Ekkor a teljes jellemzőkészletből elhagytuk az egyes jellemzőcsoportokat, majd a maradék jellemzőkre támaszkodva tanítottunk. Ennek eredményei a 2. táblázatban találhatóak. Az adatok azt mutatják, hogy az adott jellemzőcsoportot elhagyva hogyan változott az eredmény. A csökkenő (negatív) eredmény azt jelzi, hogy a vizsgált jellemzőcsoportnak pozitív hatása van az esemény felismerésben.

Table 2. A porlasztásos mérés eredményei (%)

Elhagyott jellemzők	Változás az F-mértékben
Felszíni jellemzők	**-0,28**
Morfológiai jellemzők-1	**-2,51**
Morfológiai jellemzők-2	**-0,52**
Morfológiai jellemzők-3	**-2,01**
Elemzőfa jellemzők-1	**-1,92**
Elemzőfa jellemzők-2	**-0,52**
Szózsák jellemzők-1	**-1,34**
Szózsák jellemzők-2	**-2,42**
Szózsák jellemzők-3	**-0,57**
WordNet jellemzők-1	**-0,32**
WordNet jellemzők-2	**-6,51**
WordNet jellemzők-3	**-0,53**
WordNet jellemzők-4	**-0,2**
Lista jellemzők	**0.0**
Kombinált jellemzők - 2 eleműek	**-0,79**
Kombinált jellemzők - 3 eleműek	**+0,1**

Ha a hasonló jellemzőcsoportokat összevonjuk, akkor a következő eredményeket kapjuk az összevont csoportokra (3. táblázat):

Table 3. A porlasztásos mérés eredményei - összevonással (%)

Elhagyott jellemzők	Változás az F-mértékben
Morfológiai jellemzők	**-1,63**
Szózsák jellemzők	**-4,0**
Elemzőfa jellemző	**-1,56**
WordNet jellemző	**-7,7**
Kombinált jellemzők	**-0,95**

A 2. és a 3. táblázat eredményein látszik, hogy majdnem minden jellemzőcsoportnak pozitív hatása volt a modell teljesítményére. Legjobb hatása a *WordNet* és a *Szó-*

zsák jellemzőknek volt, de sokat javítottak a *Morfológiai* és az *Elemzőfa* jellemzők is. Mindkét morfológiai elemző hatása pozitív volt. A WordNet jellemzők-2 részcsoportnak volt a legjobb hatása (6.51%). Ebben használtuk együtt a WordNet-et a szózsák modellel. A Lista jellemzőknek nem volt hatása. Negatív hatása volt a 3 elemű kombinált jellemzőknek, de a 2 elemű kombinált jellemzők hasznosak voltak.

A modellünk, amelynek eredményét a 6.2-es fejezetben ismertettünk, már csak a pozitív hatású jellemzőket tartalmazta.

6.4 Az eredmények összehasonlítása a kapcsolódó munkákkal.

Angol szövegekre Jeong és társa [6] 71,8%-os, Romeo és társai [12] 67%-os F-mértéket értek el. Olasz nyelvre Caselli [3] 69%-os, spanyol nyelvre Peris és társai [11] 59,6%-os F mértéket értek el. A kapcsolódó munkákkal összehasonlítva, eredményeink (F-mérték = 71,9%) jónak számítanak.

Összegzés

Munkánkban bemutattunk gazdag jellemzőtéren alapuló gépi tanuló megközelítésünket, amely automatikusan képes magyar nyelvű szövegekben főnévi eseményeket detektálni. Öt részterületet vizsgáltunk meg, összesen 10 000 mondattal. Gazdag jellemzőtéren alapuló *jellemzőkészletünkben* felszíni, morfológiai, függőségi elemzőfa, szózsák, Wordnet, lista és kombinált jellemzőket használtunk fel. Ezen jellemzőcsoportok mellet kiegészítő módszereket is alkalmaztunk, amelyek javították modellünk hatékonyságát, valamint a futási időt. Algoritmusainkat tesztadatbázisokon kiértékelve, versenyképes eredményeket érnek el az eddig bemutatott angol és más nyelvű eredményekkel összehasonlítva.

Bibliográfia

1. Bethard, S., Martin, J.H.: Identification of event mentions and their semantic class. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 146–154. Association for Computational Linguistics (2006)
2. Boguraev, B., Ando, R.K.: Effective use of Timebank for TimeML analysis. In: Schilder, F., Katz, G., Pustejovsky, J. (eds.) Annotating, Extracting and Reasoning about Time and Events. LNCS, vol. 4795, pp. 41–58. Springer, Heidelberg (2007)
3. Caselli, T., Russo, I., Rubino, F.: Recognizing deverbal events in context. In: Proceedings of CICLing 2011, poster session. Springer (2011)
4. Csendes, D., Csirik, J.,Gyimóthy, T.: The Szeged corpus: a POS tagged and syntactically annotated hungarian natural language corpus. In: Sojka, P., Kopecek, I., Pala, K. (eds.) TSD 2004. LNCS (LNAI), vol. 3206, pp. 41–47. Springer, Heidelberg (2004)
5. Gorzitze, S., Pado, S.: Corpus-based acquisition of German event- and object denoting nouns. In: Proceedings of KONVENS 2012 (Main Track: Poster Presentations), pp. 259–263 (2012)
6. Jeong, Y., Myaeng, S.: Using syntactic dependencies and Wordnet classes for noun event recognition. In: The 2nd Workhop on Detection, Representation, and Exploitation of Events

in the Semantic Web in Conjunction with the 11th International Semantic Web Conference, pp. 41–50 (2012)
7. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice-Hall, Upper Saddle River (2000)
8. Lesk, M. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation, pp. 24-26, New York, NY, USA. ACM. (1986)
9. Llorens, H., Saquete, E., Navarro-Colorado, B.: TimeML Events recognition and classification: learning CRF models with semantic roles. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 725–733. Association for Computational Linguistics (2010)
10. Miháltz,M., Hatvani, Cs., Kuti, J., Szarvas, Gy., Csirik, J., Prószéky, G., Váradi, T.: Methods and results of the Hungarian WordNet project. In: Tanács, A., Csendes, D., Vincze,V., Fellbaum, C.,Vossen,P., (eds.) Proceedings of the Fourth Global WordNet Conference (GWC 2008), pp. 311–320. University of Szeged, Szeged (2008)
11. Peris, A.,Taule, M.,Boleda, G.,Rodriguez,H.: ADN-classifier: automatically assigning denotation types to nominalizations. In: Proceedings of the Seventh LREC Conference, 19-21 May 2010, Valetta, Malta, pp. 1422–1428 (2010)
12. Romeo, L.,Lebani, G.E.,Bel,N., Lenci, A.: Choosing which to use? A study of distributional models for nominal lexical semantic classification. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014), pp. 4366–4373 (2014)
13. Sauri, R., Knippen,R., Verhagen, M.,Pustejovsky, J.: Evita: a robust event recognizer for QA systems. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 700–707. Association for Computational Linguistics (2005)
14. Subecz, Z.: Detection and classification of events in Hungarian natural language texts. In: Sojka, P., Horak, A., Kopecek, I., Pala, K. (eds.) TSD 2014. LNCS (LNAI), vol. 8655, pp. 68–75. Springer, Heidelberg (2014)
15. Tron, V., Kornai, A., Gyepesi, G., Németh, L., Halácsy, P., Varga, D. Hunmorph: Open source word analysis. In Proceedings of the Workshop on Software, Software '05, pp. 77–85, Stroudsburg, PA, USA. Association for Computational Linguistics. (2005)
16. Zsibrita, J., Vincze,V., Farkas,R.: magyarlanc: a toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP 2013, pp. 763–771 (2013)

A Dologfelismerő

Novák Attila[1,2], Siklósi Borbála[2]

[1] MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport ,
[2] Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar ,
1083 Budapest, Práter utca 50/a
e-mail:{novak.attila.siklosi.borbala}@itk.ppke.hu

Kivonat A szóbeágyazási modellek megjelenése az utóbbi években forradalmi változásokat hozott a nyelvtechnológia számos területén. A tömör valós vektorokkal való jelentésreprezentáció ugyanakkor közvetlenül nem interpretálható az emberek számára, bár a különböző vizualizációs technikák segítenek a modellek értelmezésében. Jelen cikkben egy olyan technikát mutatunk be, amely a szavakhoz diszkrét szemantikai jegyeket rendelve segíti a folytonos modellben ábrázolt jelentések értelmezését, ugyanakkor hozzáférhetővé teszi az azokban reprezentált tudást a diszkrét jegyekkel dolgozó gépi tanuló vagy keresőalgoritmusok számára is. Kísérleteink során hasonló magyar nyelvű erőforrások hiányában angol nyelvű lexikai erőforrásokban szereplő kategóriacímkéket rendeltünk magyar szavakhoz. Az alkalmazott transzformációk ellenére a modell jól címkézi a gyakori szavak mellett a semmilyen kézzel készített erőforrásban nem szereplő ritka, szleng szavakat, a neveket és a rövidítéseket is.

1. Bevezetés

A szavak disztribúciós viselkedésének reprezentálására az utóbbi években egyre népszerűbbé vált szóbeágyazási modellek igen hatékony módszernek bizonyultak [13]. Azonban a szavak sokdimenziós térben való absztrakt reprezentációja az emberek számára önmagában nehezen értelmezhető. Ennek orvoslására egy olyan módszert mutatunk be, ami egy nagy korpuszból létrehozott szóbeágyazási modellhez olyan szemantikai kategóriacímkéket tesz hozzá, amik az eredeti modell értelmezését segítik. A hozzáadott címkéket létező lexikai erőforrásokból, azok automatikus transzformációjával illesztjük az eredeti beágyazási térbe. Ennek köszönhetően az eredetileg nagyon sok szót tartalmazó szemantikai tér felbontható jóval kisebb számú, címkézett altérre, ami a modellt átláthatóbbá teszi.

A bemutatott algoritmus az eredeti korpuszban lévő összes szóhoz képes kategóriacímkét rendelni, függetlenül attól, hogy az adott szóalak a címkék létrehozásához használt lexikai erőforrásban szerepelt-e. Továbbá, a módszer nyelvfüggetlen, a felhasznált erőforrások nyelve nem szükségszerűen azonos az eredeti szóbeágyazási modell nyelvével. A szavak kategorizálásakor a szóbeágyazási modellek természetéből adódóan nem egy előre definiált tudományos rendszertani besorolás érvényesül, hanem a szavak reprezentációjának alapja azok disztribúciós viselkedése, tehát a tényleges nyelvhasználat.

A módszert magyar nyelvre mutatjuk be, de más nyelvre is könnyen adaptálható, amennyiben egy szóbeágyazási modell (vagy egy elégséges méretű korpusz) rendelkezésre áll.

2. Kapcsolódó munkák

A szóbeágyazási modellek az utóbbi évek egyik legnépszerűbb eszköze a szavak jelentésének hatékony reprezentációjára [12,19]. Akár szó-, akár jelentésbeágyazásról van szó, az ezekben használt folytonos vektor reprezentációk nem alkalmasak az emberi értelmezésre. Történtek kísérletek ezeknek a beágyazási modelleknek olyan létező szemantikai erőforrásoknak az „összeházasítására", mint a BabelNet [18] vagy a WordNet [6,1]. Rothe és Schütze a szóbeágyazási vektorok kombinálásával próbált WordNet synseteket belevetíteni az eredeti beágyazási térbe [20]. Más megközelítések pedig kézzel annotált adathalmazhoz próbálták adaptálni az eredeti modellt [9]. Több kutatás során pedig tudásbázisok felhasználásával próbálták javítani a beágyazási modellek minőségét [25,2,4]

3. Lexikai erőforrások

Az eredeti modellhez hozzárendelt kategóriacímkéket létező, angol nyelvű lexikai erőforrásokból nyertük ki.

Angol nyelvre az egyik legnépszerűbb ilyen erőforrás a **WordNet** [7,14], ahol a fogalmak egy szigorú hierarchikus rendszerbe vannak besorolva. A probléma azonban ezzel az erőforrással az, hogy egyrészt az alacsonyabb szinteken igen nagy a felbontása, a magasabb szintű kategóriák viszont túl általánosak [3]. Másrészt, a középső szinteken olyan mesterkélt kategóriákat tartalmaz, ami egy tudományos taxonómiában indokolt, azonban a mindennapi nyelvhasználatnak nem feltétlenül része (pl. *páros ujjú patás*). Ráadásul a WordNetben a synseteknek nincs neve sem, csak azonosítója és definíciója. Ezért a WordNetet végül nem használtuk az itt leírt kísérleteinkben.

Egy másik, gyakran használt, bár kissé elavult lexikai erőforrás a **Roget's Thesaurus** [5]. A digitálisan elérhető változata 990 szemantikai kategóriát tartalmaz. Minden kategória alatt 5 szófaj szerinti bontásban (főnév, ige, melléknév, határozószó, kifejezés/indulatszó) az adott kategória/szófaj alá sorolható szavak listája található. Az eredeti tezaurusz 91608 szót, illetve kifejezést tartalmaz, azonban a kísérleteink során használt az angol Wikipédiából épített szóbeágyazási modellünkben ezekből csak 51108 szó szerepel.[3] Mivel a modellben csak szavak szerepelnek, ezért a két halmaz metszetéből hiányoznak a többszavas kifejezések, az elavult szavak, illetve a téves szófajcímkével ellátott szavak.

Szintén online elérhető a **Longman Dictionary of Contemporary English** (LDOCE) [23] digitális változata, amiből könnyen előállítható egy az előző erőforráshoz hasonló gyűjtemény is, hiszen a benne lévő címszavak egy része 213

[3] Hogy a Wikipédiából épített modellt hogyan építettük, és pontosan mire és hogyan használtuk, az a 4. részben fog kiderülni.

szemantikai kategóriába van besorolva, szintén szófaj szerinti bontással együtt. Mivel azonban ez a szótár jóval modernebb szókincset tartalmaz, ezért az angol Wikipédia-modellel való metszés után az eredeti 28257 kategorizált szóból 21546 megmaradt.

A harmadik erőforrás a szintén az LDOCE-n alapuló **4lang** szótár volt. Ebben az erőforrásban az eredeti szótár definícióinak formális átirata szerepel [8], ahogy az az alábbi példán látszik:

```
bread: food, FROM/2742 flour, bake MAKE
show: =AGT CAUSE[=DAT LOOK =PAT], communicate
```

Ezt az ábrázolásmódot tovább alakítottuk úgy, hogy az előzőeknek megfelelő formát kapjunk. Ehhez a formális definíciókat feldaraboltuk (szóközök és zárójelek mentén) és minden egyes így kapott darabot címkének tekintettünk, összegyűjtve hozzá minden olyan szót, aminek a leírásában az adott címke szerepelt. Így 1489 kategóriacímke jött létre ebből a szótárból, amelyek összesen 12507 szóval voltak összerendelve. Ezekből a Wikipédia-szókinccsel való metszés után 11039 szó maradt. Annak ellenére, hogy ebben a szótárban főleg gyakori szavak szerepelnek, mégis voltak olyan elemek, amik nem szerepeltek az elemzett Wikipédiából készített modellben. Ezek többnyire toldalékok, illetve toldalékolt szóalakok.

Az 1. táblázatban néhány példa látható a kategóriacímkékre és a hozzájuk rendelt szavakra.

Erőforrás	**Kategória**	**Példák az eredeti erőforrásból**
ROGET	Mean_N	medium#NN generality#NN neutrality#NN middle_state#NN median#NN golden_mean#NN middle#NN
ROGET	Rotundity_ADJ	spherical#JJ cylindric#JJ round_as_an_apple#JJ bell_shaped#JJ spheroidal#JJ conical#JJ globated#JJ
LDOCE	Cooking	allspice#NN bake#VB barbecue#VB baste#VB blanch#VB boil#VB bottle#VB bouillon_cube#NN
LDOCE	Mythology	centaur#NN chimera#NN Cyclops#NN deity#NN demigod#NN faun#NN god#NN griffin#NN gryphon#NN
4LANG	food	sandwich#NN, fat#NN, bread#NN, pepper#NN, meal#NN, fork#NN, egg#NN, bowl#NN, salt#NN
4LANG	=DAT	say#VB, show#VB, allow#VB, swear#VB, grateful#ADV, let#VB, teach#VB, give#VB, help#VB

1. táblázat. Példák az eredeti lexikonokban található kategóriákból és hozzájuk tartozó szavakból azok azonos formára való átalakítása után

A 2. táblázat első négy oszlopa foglalja össze a felhasznált erőforrások jellemzőit. A 4.2 részben leírt módon az angol Wikipédiából épített modell alapján klaszterezést is végeztünk az egyes kategóriákat jellemző szavakon. Ennek eredménye látható az utolsó három oszlopban.

4. Módszer

Célunk egy olyan eszköz létrehozása volt, ami egy tetszőleges szóhoz hozzárendeli a megfelelő szemantikai kategóriacímkéket, akkor is, ha az adott célszó nincs benne egyik lexikai erőforrásban sem, illetve, ha ilyen lexikai erőforrás az adott nyelven nem is létezik. Ezért két problémát kellett kezelni: a kategóriacímkék hozzárendelését és a nyelvi különbség áthidalását.

Erőforrás	Eredeti			Metszet és klaszterezés után		
	kategória	szó	szó/kat.	kategória	szó	szó/kat.
LDOCE	213	28257	132,66	3069	21546	7,02
ROGET	3077	91608	29,77	7066	51108	7,23
4LANG	1489	12507	8,39	2249	11039	4,91

2. táblázat. A felhasznált erőforrások jellemzői (különböző kategóriák száma, szavak száma, átlagos szószám kategóriánként; az angol modellel való metszés, illetve a klaszterezés előtt és után).

4.1. Szóbeágyazási modellek létrehozása

A nyelvtechnológiai kutatások egyik kurrens módszere a folytonos vektoros (*word embedding*) reprezentációk alkalmazása, melyek nyers szöveges korpuszból szemantikai információk kinyerésére alkalmazhatók. Ebben a rendszerben a lexikai elemek egy valós vektortér egyes pontjai, melyek konzisztensen helyezkednek el az adott térben. A módszer hátránya csupán az, hogy önmagában nem képes a poliszémia, illetve homonímia kezelésére, tehát egy többjelentésű lexikai elemhez is csupán egyetlen jelentésvektort rendel. Ennek részleges kezelésére egy egyszerű megoldást alkalmaztunk azokban az esetekben, ahol az azonos alakok különböző szófajúak. Ehhez a modell építése előtt szófaji egyértelműsítést és lemmatizálást alkalmaztunk a korpuszra[4] a PurePos szófaji egyértelműsítő [17] és a Humor morfológiai elemző [15,16] használatával, majd a fő szófajcímkéket hozzáfűztük a szótövekhez, így az azonos alakú, de különböző szófajú szavaknak külön reprezentációja jött létre. Korábban azt is megmutattuk, hogy az összetett morfológiájú nyelvek esetén jobb minőségű szóbeágyazási modell hozható létre, ha a további morfológiai címkékben kódolt információk különálló tokenként maradnak meg a modell építéséhez használt szövegben, így a hozzájuk tartozó szótő kontextusában jelennek meg [22,21].

Mivel a jelen cikkben bemutatott címkézőrendszer megvalósítása során a magyar modellt egy angol szóbeágyazási modellnek is meg akartuk feleltetni, ezért ezt is hasonló módon hoztuk létre. A 2,25 milliárd szavas angol Wikipédia[5] szövegeit a Stanford tagger [24] használatával elemeztük, a szófajcímkéket a szótövekhez csatoltuk, a további morfológiai címkéket pedig külön tokenként leválasztottuk.

Mind az angol, mind a magyar modell tanításához a word2vec[6] eszközben implementált CBOW modellt használtuk, 5 token sugarú szövegkörnyezetet véve figyelembe és 300 dimenziós beágyazási modelleket hozva létre.

4.2. Szemantikus kategóriacímkék beágyazása

Ha van egy beágyazási modellünk, akkor az abban szereplő szóvektorok klaszterezésével könnyen létrehozható egy az eredeti szótárnál kevesebb elemből álló reprezentáció, amiben az egyes klaszterekbe tartozó szavakat valamilyen szempont szerint hasonló szemantikai jegyekkel rendelkező szavaknak tekinthetjük.

[4] A korábbi modelljeink [22,21] építéséhez használt webkorpuszt alkalmaztuk itt is.
[5] letöltve: https://dumps.wikimedia.org/ 2016. május
[6] https://code.google.com/p/word2vec/

Ebben az esetben azonban ezeknek a közös szemantikai jegyeknek a meghatározása csupán kézzel lehetséges, de még akkor is nehézséget jelenthet ezeknek a csoportoknak a felcímkézése ember által értelmezhető formában. Továbbá, hacsak nem valamilyen probabilisztikus klaszterezést alkalmazunk, minden szó csak egy klaszterbe kerül.

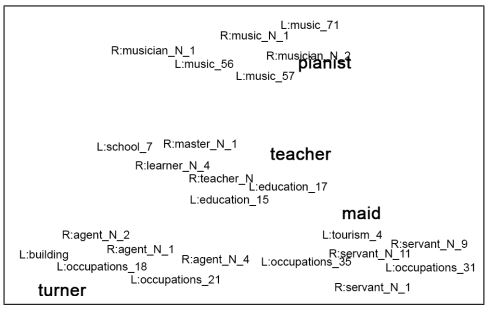
Az elnevezés problémája kezelhető lenne úgy, hogy a klaszter egy reprezentatív elemét (pl. a centroidhoz legközelebbit) kiválasztjuk, azonban ebben az esetben nem a csoportba tartozó szavak közös szemantikai kategóriáját határoznánk meg, hanem csupán a kategória egy példányát jelölnénk meg. A probléma megoldására tehát azt a módszert alkalmaztuk, hogy a fent felsorolt lexikai erőforrások kategóriacímkéit helyeztük el az eredeti szóbeágyazási modell által létrehozott szemantikai térben.

Az eredeti erőforrásokban szereplő kategóriacímkék azonban néha túl általánosnak bizonyultak, ezért a hozzájuk rendelt szólista is igen heterogén volt. Ezért először az eredeti kategorizációt tovább bontottuk úgy, hogy egy hierarchikus klaszterezési algoritmus [22] segítségével csoportosítottuk az 5-nél több szót tartalmazó kategóriacímkéhez tartozó szólistákat. Ennek eredményeképpen minden ilyen címkéhez alcsoportok jöttek létre, melyeket numerikus indexszel különböztettünk meg egymástól. A 2. táblázat utolsó három oszlopában láthatók a klaszterezés eredményeként kapott lexikonok jellemzői.

A klaszterezés során a Roget's Thesaurus elavult szóhasználatából adódó problémát is sikerült némileg áthidalni. Mivel az egyes szavakhoz tartozó reprezentációt a klaszterezéshez a modern Wikipédiából épített modell alapján nyertük ki, ezért az esetleg korábban más jelentéssel bíró szóalakokhoz is azok modern jelentését tudtuk reprezentálni. Például a *Combatant* kategóriába tartozó szavak közül a *charger, battery, file, monitor* külön klaszterbe került, hiszen ezek ma már inkább számítástechnikai/elektronikai jelentést hordoznak. Így bár maga a kategóriacímke nem feltétlenül jellemzi jól a hozzá tartozó szemantikai jegyet, de a klaszterezés során hozzáadott numerikus index alapján azonosítható és jól elválasztható ez a kategória a *Combatant* címkéhez tartozó szavakból létrejött többi, katonai kifejezéseket tartalmazó kategóriától.

Ezután minden így létrejött új címkéhez hozzárendeltük a benne felsorolt szavak beágyazási vektorának átlagát a szófajcímkékkel ellátott és tövesített angol Wikipédia-modellből. Így megkaptuk a kategóriacímkék pozícióját az angol szóbeágyazási térben. A címkékhez tartozó vektorokat külön tároltuk, hogy a lekérdezés során könnyen le lehessen szűkíteni az eredményt kategóriacímkékre, illetve szavakra. Egy angol, szófajcímkével ellátott szóhoz tehát úgy kaphatjuk meg a megfelelő kategóriacímkéket, hogy az őt reprezentáló vektorhoz koszinusztávolság alapján legközelebbi vektorokat kérdezzük le a kiválasztott erőforrásban szereplő kategóriacímkék vektorai közül .

Az 1. ábrán négy angol szó (*pianist 'zongorista', teacher 'tanár', turner 'esztergályos', maid 'takarítónő'*) és a LDOCE és Roget modellekből a hozzájuk tartozó 3 legközelebbi kategóriacímke elhelyezkedése látható két dimenzióba leképezve.



1. ábra. A *pianist, teacher, turner, maid* szavakhoz a LDOCE és a Roget modellekből lekérdezett 3-3 legközelebbi címke elhelyezkedése a szemantikai térben

4.3. Nyelvek közötti leképezés

Korábbi kutatások bemutatták, hogy a különböző nyelvekre létrehozott szóbeágyazási modellek által definiált szemantikai terek leképezhetők egymásba a leképezés során egy kiindulási szótár alapján megtanult páronkénti lineáris transzformáció alkalmazásával [11]. Ha a kiindulási szótárban n darab (w_x, w_z) szópár van, ahol w_x fordítása w_z, a vektorreprezentációik pedig $(x_i, z_i)_{i=1}^n$, akkor a W transzformációs mátrix az alábbi optimalizációs probléma megoldásaként meghatározható:

$$\min_W \sum_{i=1}^{n} ||Wx_i - z_i||^2 \tag{1}$$

Így minden forrásnyelvi x vektorra alkalmazható a $z = Wx$ transzformáció. A célnyelvi modellben a z vektorhoz közel található szavak x közelítő fordításai.

A 4lang-szótár 3477 angol-magyar szópárból álló részhalmazát használtuk fel a transzformációs mátrix tanításához szükséges kiinduló szótárként. Az eredeti szótárból azokat a párokat tartottuk meg, amiknek mindkét tagja legalább 10000-szer előfordul a megfelelő nyelvű korpuszban. További 100 szópáron kézzel kiértékelve a transzformáció 38%-os pontosságot adott az első legközelebbi szóra nézve, és 69%/81%-ot az első 5/10 legközelebbi szóra nézve. Ez azt jelenti, hogy a transzformáció során a megfelelő környékre történik a leképezés az esetek nagy részében. Mivel célunk nem a pontos fordítások azonosítása volt, hanem a magyar és az angol nyelvű szemantikai tér egymásra illesztése, ezért ez az eredmény igazolta a transzformáció alkalmazhatóságát. Ez a módszer tehát lehetővé teszi, hogy az angol erőforrásokból létrehozott szemantikai kategóriacímkékhez rendelt vektorokat az angol térből leképezzük a magyar nyelvű szóbeágyazási modell terébe.

5. Kísérletek és eredmények

A módszerünk elsődleges célja a szóbeágyazási modellek értelmezhetőségének támogatása, illetve a beágyazási tér különböző részeinek szemantikai jegyekkel való

automatikus annotálása. Ezért először az eredmények kiértékeléséhez egy webes felületbe integrált ábrázolásmódot használtunk, ami a t-sne algoritmus [10] alkalmazásával az eredetileg 300 dimenziós beágyazási teret kétdimenziós ábrán jeleníti meg. A felület lehetőséget ad arra, hogy magyar szavakhoz bármely modellből (Roget's, LDOCE, 4lang) lekérdezhessünk tetszőleges számú kategóriacímként és az így kapott annotált szemantikai teret megjelenítsük. A vizualizáció mellett azonban kvantitatív kiértékelést is végeztünk, különböző típusú szavakra vizsgálva a kategorizáció minőségét.

5.1. Általános szavak

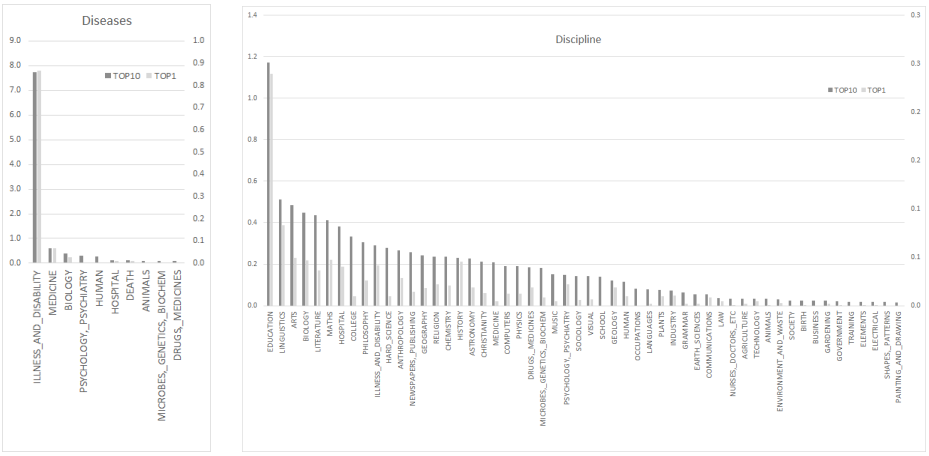
A sztenderd szóalakok kiértékeléséhez különböző szemantikai csoportokba sorolható szavakat gyűjtöttünk össze a [22]-ben bemutatott félautomatikus módszerrel, majd kézzel ellenőriztünk 35 ilyen csoportot (3. táblázat), amiben összesen 50507 szó szerepelt. Így ezeket gold standardnek tekintettük.

Csoport	**Szavak száma**	**Csoport**	**Szavak száma**
Foglalkozások	1332	Épületek	1123
Tudományágak	1051	Idő	380
Mértékegységek	1229	Esemény	3538
Elektronika	935	Színek	907
Betegségek	1359	Ruhák jellemzői	432
Állatok	1189	Emberek jelzői	980
Konyhai eszközök	769	Ételek jelzői	1141
Ételek	1662	Mozgást jelentő igék	1113
Járművek	1084	Szabadidős tevékenységek	1953
Ruhák	915	Pusztulást jelentő igék	909
Vizek	881	Magyar vezetéknevek	4197
Területek	1313	Latin vezetéknevek	2752
Természeti események	643	Angol vezetéknevek	1738
Domborzat	678	Szláv nevek	1479
Városok	4568	Becenevek	537
Helyek	3295	Emberi kapcsolatok	959
Embercsoportok	1206	Köszönések	322
Sportolók	445	**Sum**	**50507**

3. táblázat. A kiértékeléshez használt kézzel ellenőrzött szócsoportok és azok mérete

Ezután minden szóhoz mindhárom modellből lekérdeztük a 10 legközelebbi kategóriacímkét, majd összeszámoltuk, hogy az egyes csoportokon belül melyik címke hányszor fordult elő (függetlenül attól, hogy hányadik helyen szerepelt a 10-es listában). Egy másik esetben pedig csak az első helyen szereplő címkéket számoltuk össze, aminek célja a módszer pontosságának kiértékelése volt, azonban a tágabb értelmű csoportok esetén értelmetlennek bizonyult csupán egyetlen címke hozzárendelése, így az ebben az esetben mért alacsonyabb pontosság értékek nem feltétlenül jelentenek rosszabb teljesítményt.

A 2. ábra a *betegségek* és a *tudományágak* csoportjaira kapott eredményt mutatja a LDOCE címkék hozzárendelésére vonatkozóan. A grafikon egyes oszlopai a csoportban szereplő azon szavak arányát jelölik, amikhez az adott címkét rendelte a rendszer (az első 10 címke közül bármelyik pozícióban, illetve első címkeként). A címkékre vonatkozó összesítést a hozzájuk a klaszterezés során rendelt



(a) Betegségek (b) Tudományágak

2. ábra. A *betegségek* és a *tudományágak* csoportjába tartozó szavakhoz rendelt első 10 (TOP10) és az első (TOP1) címke eloszlása

index elhagyásával számoltuk, így ugyanaz a főcímke egy szó első 10 címkéjének listájában többször is előfordulhat, ezért jelennek meg 1-nél nagyobb aránynak megfelelő értékek az ábrán. Annak ellenére, hogy az összesítésben elhagytuk ezeket az indexeket, az indexek által reprezentált különbségek jelentősek lehetnek. Például a *biology* címkén belül elválnak a betegségekre, az emberi szervekre vagy a sejtbiológiára vonatkozó címkék, azonban ezek a csupán számokkal jelölt különbségek az emberi értelmezést (ilyen formában) nem segítik (egy gépi tanulási algoritmusban való felhasználás során azonban mindenképpen érdemes ezeket is figyelembe venni).

Látható, hogy a betegségeket tartalmazó csoport esetén a leggyakoribb címke az 'ILLNESS AND DISABILITY', ami az elsőként hozzárendelt címkék 78%-a, ami mellett csak néhány további címke jelenik meg számottevő arányban (*medicine; biology; psychology, psychiatry; human; hospital; death; animals; stb.*). A tudományágak csoportjában azonban sokkal kevésbé meredek a címkék eloszlásának íve. Bár a leggyakoribb címke ('EDUCATION') itt is kiemelkedik a többi közül és általánosságban jellemzi a csoportot, ezt a különböző tudományágak nevei követik közel egyenletesen csökkenő eloszlás szerint (*linguistics, arts, biology, literature, maths, etc.*). Látható tehát, hogy csoportonként eltérő lehet a címkék eloszlásának jellege, ezért a kiértékelés során nem a tényleges minőséget jellemezte volna az összes csoportra vonatkozó összesített eredmény. A 4. táblázat további néhány csoport szavaihoz az első 10 közül bármelyik helyen leggyakrabban előforduló kategóriacímkéket tartalmazza az egyes modellekből (L: LDOCE, R: Roget, F: 4lang). A példaként felsorolt csoportoknál látható az is, hogy a negyedik oszlopban szereplő összes különböző kategóriacímkék száma a csoport heterogenitásától függően eltérő, ugyanakkor minden esetben jóval kisebb, mint a csoportban szereplő szavak száma, tehát elmondható, hogy az alkalma-

Csoport	szavak	TOP 10 kategóriacímke	D	COV
Foglalkozások	1332	L: occupations, education, college, newspapers, publishing, painting and drawing, nurses, doctors, etc, music, construction, building, literature	80	71.25%
		R: Agent_N, Scholar_N, Remedy_N, Artist_N, Experiment_N, Book_N, Servant_N, Clothing_N, Painting_N, Accounts_N	97	61.26%
		F: HAS, profession, person, skill, scientist, educate, job, practice, science, IN/2758	112	67.64%
Mértékegységek	1229	L: measurement, currencies, computers, electricity, broadcasting, drink, maths, jewellery, numbers, elements	41	91.70%
		R: Length_N, Money_N, Gravity_N, Receptacle_N, Greatness_N, Littleness_N, Smallness_N, Period_N, Calefaction_N, Heat_N	64	89.50%
		F: unit, length, HAS, measure, =REL, temperature, cent, small, pound, mass	105	77.14%
Betegségek	1359	L: illness_and_disability, medicine, biology, psychology, psychiatry, human, hospital, death, animals, microbes, genetics, biochem, drugs, medicines	15	99.19%
		R: Disease_N, Death_N, Deterioration_N, Agitation_N, Hindrance_N, Disease_ADJ, Convexity_N, Remedy_N, Violence_N, Evil_N	35	91.83%
		F: bad, health, body, ill, disease, organ, situation, injury, damage, harm	37	89.92%
Természeti események	643	L: nature, meteorology, geography, illness_and_disability, geology, physics, earth_sciences, chronology, astronomy, power	41	80.87%
		R: River_N, Wind_N, Disease_N, Violence_N, Deterioration_N, Revolution_N, Evil_N, Agitation_N, Rotation_N, Resentment_N	88	57.39%
		F: cloud, wind, weather, ice, IN/2758, atmosphere, sudden, damage, AT/2744, HAS	92	55.99%
Emberek jelzői	980	L: animals, hair_and_beauty, clothes, colours, occupations, illness_and_disability, nature, clothes_and_fashion, biology, psychology, psychiatry	66	67.55%
		R: Size_ADJ, Clothing_N, Love_ADJ, Beauty_ADJ, Adolescence_ADJ, Animal_N, Sexuality_ADJ, Servant_N, Vulgarity_ADJ, Pleasurableness_ADJ	159	49.90%
		F: HAS, lack, kind, CAUSE, mad, IN/2758, bad, much, intelligent, body	100	66.94%
Mozgást jelentő igék	1113	L: transport, air, computers, theatre, swimming, government, water, insects, illness_and_disability, motor_vehicles	48	73.76%
		R: Journey_VB, Velocity_VB, Arrival_VB, Depression_VB, Navigation_VB, Departure_VB, Ascent_VB, Supposition_VB, Offer_VB, Haste_VB	81	71.70%
		F: =AGT, after, lack, AT/2744, go, =PAT, surface, rush, long, ON	48	62.71%

4. táblázat. Néhány kategóriához rendelt leggyakoribb címkék a három modellből. D=különböző címkék száma, COV=az első 10 címke aránya az összes hozzárendelt címkéhez képest

zott módszerrel hatékonyan sikerült az eredeti szóbeágyazás által meghatározott szemantikai térben szereplő sűrű numerikus vektorokat emberi értelmezésre is alkalmas szimbolikus jellemzők egy korlátozott méretű halmazára leképezni.

Bár a LDOCE címkék elnevezései a legérthetőbbek, a Roget és 4lang modellek alapján is hasznos szemantikai jegyeket határoztunk meg. Míg például a Roget modellben a melléknevek kategorizációja sokkal kifinomultabb, a 4lang szótárból kinyert címkék másfajta értelmezést rendelnek a szavakhoz. Mivel ebben az esetben a címkék a szótárban szereplő definíciók részei, néhánynak önmagában nincs valódi jelentéstartalma (pl. HAS), viszont a szavakhoz rendelt 10 legközelebbi címkét együttesen vizsgálva a szótárban eredetileg nem szereplő szavakhoz is egy definíció-szerű leírást adnak meg.

A címkézés további jellemzője, hogy mivel a szavak reprezentációja a tényleges nyelvhasználat alapján jött létre, ezért a rendszer kategóriákat is ehhez a fajta reprezentációhoz rendel, nem pedig egy előre definiált tudományos rendszerezés szerint. Tehát például a *macska* szónak több közös címkéje van a *kutya* szóval, mint az *oroszlán* vagy *tigris* szavakkal. Egy biológiai rendszertan természetesen a macskaféléket tekinti közelebbi rokonoknak, azonban a mindennapi életben a háziállat–vadállat megkülönböztetés sokkal jellemzőbb.

5.2. Tulajdonnevek

A szóbeágyazási modellek a létrehozásukhoz használt korpuszban implicit megtalálható világismeretet is hatékonyan tükrözik. Ezért a kategóriacímke-hozzárendelés különböző típusú tulajdonnevek, vagy akár rövidítések esetén is működik. Az

5. táblázatban látható, hogy személynevekhez is releváns címkéket rendel, még akkor is, ha az adott név nem feltétlenül gyakori, de egyértelműen azonosítható. Hasonlóan jól működik a hozzárendelés a különböző szervezetek, intézmények rövidített nevei esetén, ahol még az állami és egyházi oktatási intézmények közötti különbség is megjelenik az *ELTE*, illetve a *PPKE* címkehalmazaiban.

Látható tehát, hogy a módszerünk az olyan szavakhoz is releváns címkéket rendel, amik sem a felhasznált lexikai erőforrásokban, sem az angol szóbeágyazási modellben nem szerepeltek. Az is látszik ezekből az eredményekből, hogy a többszörös transzformáció során sem veszett vagy torzult el a lényegi szemantikai információ jelentős része.

Szó	**TOP 10 kategóriacímke**
Bartók	L: MUSIC.20, MUSIC.71, PERFORMING.12, MUSIC.51, MUSIC.52, MUSIC.54, MUSIC.40, MUSIC.19, LITERATURE.14, MUSIC.41, MUSIC.21
	R: Music.N.5, Music.N.1, Music.N.6, Precursor.N.1, Poetry.N.3, Musician.N.2, Musician.N.5, Lamentation.N.3, Music.N.7, Music.N.9, Poetry.N.2
	F: HAS.27, music.2, art, poem, poet, poetry, WRITE, sound/993.2, text.2, musician, '.7
Obama	L: OFFICIALS.12, GOVERNMENT.17, GOVERNMENT.15, OFFICIALS.13, GOVERNMENT.18, OFFICIALS.10, GOVERNMENT.19, LAW.29, GROUPINGS.10, VOTING.7, GROUPINGS.4
	R: Government.N.14, Politics.N.2, Authority.N.4, Director.N.2, Council.N.2, Politics.N.5, Conduct.N.3, Direction.N.1, Participation.N.1, Government.N.12, Compact.N.2
	F: country.13, government, politician, HAS.22, @United.States, state/76.2, LEAD/2617, place/1026.3, president, republic, country.8
Einstein	L: HARD.SCIENCE.2, PHYSICS.1, PHILOSOPHY.1, MATHS.19, ASTRONOMY.6, LINGUISTICS.14, CHEMISTRY.22, OCCULT.1, ELECTRICITY.6, OCCUPATIONS.3, EDUCATION.14
	R: Heterodoxy.N.5, Scholar.N.2, Experiment.N.2, Smallness.ADJ.2, Intellect.N.7, Conversion.N.3, Production.N.1, Irreligion.N.1, Knowledge.N.1, Life.N.2, Irreligion.N.4
	F: @Karl.Marx, science, man/744.2, atom, scientist, poet, ABOUT.3, NOTPART.OF, prove, exact, politician
ELTE	L: COLLEGE.11, COLLEGE.13, EDUCATION.13, COLLEGE.12, EDUCATION.9, EDUCATION.10, COLLEGE.14, EDUCATION.12, SCHOOL.7, SCHOOL.2, SCHOOL.9
	R: Knowledge.N.2, School.ADJ, Language.N.1, School.N.5, Skill.N.4, Learner.N.4, Teaching.ADJ, Learner.N.3, Evidence.N.4, World.N.3, Receptacle.N.4
	F: educate, institution, study, student, degree, science, AT/2744.27, numbers, atom, GIVE.2, IN/2758.22
PPKE	L: COLLEGE.12, COLLEGE.13, EDUCATION.9, COLLEGE.8, COLLEGE.11, EDUCATION.13, EDUCATION.15, OCCUPATIONS.7, SCHOOL.9, EDUCATION.12, CHRISTIANITY.2
	R: School.ADJ, Knowledge.N.2, School.N.4, Teaching.ADJ, Churchdom.N.6, Churchdom.ADJ.1, Publication.ADJ.2, Churchdom.N.1, Skill.N.4, Evidence.N.4, Learner.N.3
	F: educate, institution, science, group.5, study, student, degree, society/2285.2, sleeve, @Catholic.Church, LEAD/2617
IBM	L: COMPUTERS.33, COMPANIES.3, PLANTS.21, COMPUTERS.34, COMPANIES.2, BUSINESS.BASICS.5, FACTORIES.3, INDUSTRY.3, COMPUTERS.62, COMMUNICATIONS.3, COMPUTERS.27
	R: Servant.N.4, Numeration.N.3, Convexity.N.14, Jurisdiction.N.2, Support.N.7, Merchant.N.3, Action.N.1, Participation.N.2, Receiving.N.2, Receptacle.N.29, Falsehood.N.3
	F: business, factory, computer, IN/2758.22, company/2549, unit.4, INSTRUMENT.5, machines, AT/2744.27, produce, method

5. táblázat. Néhány példa a hozzárendelt címkékre tulajdonnevek és rövidítések esetén a három modellből (L:LDOCE, R:Roget, F:4lang)

5.3. Szubsztenderd nyelvhasználat

Már az eredeti magyar szóbeágyazási modellben is érzékelhető volt, hogy a hasonló annotációs hibát vagy elírást tartalmazó szóalakok egymáshoz közel helyezkedtek el a modellben [22]. Bár az ilyen hibatípusok azonosítása is hasznos funkciója lehet ezeknek a modelleknek, ezek részben elfedik az azonos hibatípusba tartozó szavak közötti szemantikai különbségeket. A kategóriacímkék hozzárendelésekor azonban az ilyen hibás szóalakokhoz is helyes címkéket rendelt a modellünk.

Ugyanez igaz a szleng és más nem sztenderd szóalakokra, amik igen gyakoriak a webről gyűjtött korpusz felhasználói hozzászólásokat, fórumokat tartalmazó részében. Ráadásul ezek gyakran igen erős érzelmi töltetet is tartalmaznak. Ez jól tükröződik az olyan szavakhoz rendelt kategóriacímkékben, mint a *nyugger*, *proli*, *bolsi* vagy *cigó*, amikhez a leggyakoribb címkék például Deceiver',

‘Obstinacy’, ‘Ignorance’, Thief’, CRIME’, ‘POLITICS’ ‘RACE RELATIONS’ ‘PSYCHOLOGY, PSYCHIATRY’, stupid’, criminal’ a mindegyikre illeszkedő ‘person’ mellett.

6. Konklúzió

Bemutattunk egy olyan módszert, melynek segítségével a szóbeágyazási modellekben implicit jelen lévő jelentéscsoportokat emberek által is értelmezhető szimbolikus jegyekké transzformáltuk. A módszer olyan nyelvek esetén is alkalmazható, mint a magyar, amelyekre nem áll rendelkezésre olyan lexikai erőforrás, amelyben szereplő kategóriarendszer közvetlenül felhasználható lenne az osztályozás során. Bemutattuk, hogy egy angol szóbeágyazási modellen átvetítve sem torzul lényegesen az információ, az angol nyelvű erőforrások alapján meghatározott címkék hozzárendelése még tulajdonnevek, rövidítések, illetve a semmilyen külső erőforrásban nem szereplő nem sztenderd szóalakok esetén is jól működik.

Hivatkozások

1. Agirre, E., Martínez, D., de Lacalle, O.L., Soroa, A.: Evaluating and optimizing the parameters of an unsupervised graph-based wsd algorithm. In: Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing. pp. 89–96. TextGraphs-1 (2006)
2. Bordes, A., Weston, J., Collobert, R., Bengio, Y.: Learning structured embeddings of knowledge bases. In: AAAI (2011)
3. Brown, S.W.: Choosing sense distinctions for wsd: Psycholinguistic evidence. In: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers. pp. 249–252 (2008)
4. Camacho-Collados, J., Pilehvar, M.T., Navigli, R.: A unified multilingual semantic representation of concepts. In: Proceedings of ACL-IJCNLP 2015 – Volume 1. pp. 741–751. Association for Computational Linguistics, Beijing, China (July 2015)
5. Chapman, R.: Roget’s International Thesaurus. Harper Colophon Books
6. Chen, X., Liu, Z., Sun, M.: A unified model for word sense representation and disambiguation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1025–1035. Association for Computational Linguistics, Doha, Qatar (October 2014)
7. Fellbaum, C. (ed.): WordNet: an electronic lexical database. MIT Press (1998)
8. Kornai, A., Ács, J., Makrai, M., Nemeskey, D.M., Pajkossy, K., Recski, G.: Competence in lexical semantics. In: Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics. pp. 165–175. Association for Computational Linguistics, Denver, Colorado (June 2015)
9. Labutov, I., Lipson, H.: Re-embedding words. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics – Volume 2. pp. 489–493. Association for Computational Linguistics, Sofia, Bulgaria (August 2013)
10. van der Maaten, L., Hinton, G.E.: Visualizing high-dimensional data using t-sne. Journal of Machine Learning Research 9, 2579–2605 (2008)
11. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. CoRR abs/1309.4168 (2013)

12. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 27th Annual Conference on Neural Information Processing Systems 2013. pp. 3111–3119 (2013)
13. Mikolov, T., Yih, W., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of NAACL 2013. pp. 746–751 (2013)
14. Miller, G.A.: Wordnet: A lexical database for english. Communications of the ACM 38, 39–41 (1995)
15. Novák, A.: A new form of Humor – mapping constraint-based computational morphologies to a finite-state representation. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). European Language Resources Association (ELRA), Reykjavik, Iceland (may 2014)
16. Novák, A., Siklósi, B., Oravecz, C.: A new integrated open-source morphological analyzer for Hungarian. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA), Paris, France (may 2016)
17. Orosz, G., Novák, A.: PurePos 2.0: a hybrid tool for morphological disambiguation. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013). pp. 539–545. Hissar, Bulgaria (2013)
18. Panchenko, A.: Best of both worlds: Making word sense embeddings interpretable. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA), Paris, France (may 2016)
19. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014)
20. Rothe, S., Schütze, H.: Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In: Proceedings of ACL-IJCNLP 2015 – Volume 1. pp. 1793–1803. Association for Computational Linguistics, Beijing, China (July 2015)
21. Siklósi, B., Novák, A.: Beágyazási modellek alkalmazása lexikai kategorizációs feladatokra. XII. Magyar Számítógépes Nyelvészeti Konferencia pp. 3–14 (2016)
22. Siklósi, B.: Using embedding models for lexical categorization in morphologically rich languages. In: Gelbukh, A. (ed.) Computational Linguistics and Intelligent Text Processing: 17th International Conference, CICLing 2016. Springer International Publishing, Cham., Konya, Turkey (April 2016)
23. Summers, D.: Longman Dictionary of Contemporary English. Longman Dictionary of Contemporary English Series
24. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of NAACL 2003 - Volume 1. pp. 173–180. NAACL '03 (2003)
25. Yu, M., Dredze, M.: Improving lexical embeddings with semantic knowledge. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics – Volume 2. pp. 545–550. Association for Computational Linguistics, Baltimore, Maryland (June 2014)

Minőségbecslő rendszer egynyelvű természetes nyelvi elemzőhöz

Yang Zijian Győző[1], Laki László János[2]

[1] Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
[2] MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport
1083 Budapest, Práter utca 50/a
{yang.zijian.gyozo, laki.laszlo}@itk.ppke.hu

Kivonat A pszicholingvisztikai indíttatású természetes nyelvi elemzés egy új, emberi nyelvelemzést modellező nyelvtechnológiai módszer. Ez a modell egy valós idejű elemző, amelynek párhuzamosan több szála elemzi egyszerre a bemeneten sorban érkező szavakat, kifejezéseket vagy mondatokat. A párhuzamosan futó szálak közül az egyik a minőségbecslő modul, amely menedzseli, szűri a hibás és zajos bemenetet, valamint tájékoztatja a többi szálat a bemenet aktuális minőségéről. A minőségbecslő modul felépítéséhez a gépi fordítás kiértékeléséhez használt minőségbecslés módszerét használtuk. Ahhoz, hogy a minőségbecslő modellünk a természetes nyelvi elemző egyik párhuzamosan futó szálát képezze, ötvöztük az eredeti minőségbecslő rendszert a feladatorientált architektúrával. A kutatásunk során felépítettünk egy feladatorientált minőségbecslő rendszert, amely az egynyelvű szöveg valós idejű minőségének becslésére alkalmas. Az általunk létrehozott rendszer segítségével ∼70%-os pontossággal tudjuk megbecsülni a bemeneti szöveg minőségét. A rendszer az AnaGramma magyar nyelvű elemzőhöz készült, de más nyelvekre is használható.

Kulcsszavak: minőségbecslés, pszicholingvisztika, természetes nyelvi elemzés

1. Bevezetés

Mára a pszicholingvisztika fontos terület lett a számítógépes nyelvészetben. Amíg a hagyományos nyelvi elemzők (pl.: szintaktikai elemzők, szófaji elemzők stb.) a mondat végének elhangzása után kezdik az elemzést, addig az emberi elemző a kommunikáció során folyamatosan dolgozza fel a hallott vagy az olvasott szavakat, kifejezéseket.

Az AnaGramma [3,9] egy pszicholingvisztiaki indíttatású nyelvi elemző rendszer, amely modellálja a valós emberi nyelvi feldolgozást. Az elemző performancia alapú és szigorúan balról jobbra elemzi a bementet. A rendszer architektúrája eredendően párhuzamos. A hagyományos megközelítésekkel szemben, itt az elemzendő szót valós időben, folyamatosan dolgozza fel. A párhuzamosan jelenlévő szálak (pl.: szintaktikai elemző, morfológiai elemző, korpuszgyakorisági szálak stb.) egyszerre és egymással kommunikálva vizsgálják a bemenetet, valamint

egymás hibáit javítva végzik el az elemzést. Ezen párhuzamos szálak közül az egyik fontos szál a minőséget becslő, vizsgáló szál.

A minőségelemző szál legfontosabb feladatai, hogy minőségi jelzőket biztosít a többi szál és a felhasználó számára, valamint adott esetben kontrollálja, szűri a bemenetet.

Az AnaGramma alapvetően kétféle száltípust használ. Az egyik típus a *felkínálás* jellegű szál, amely információt ad az elemről (pl.: alanyesetű), a másik típus pedig az *igény* jellegű szál, amely egy adott tulajdonságú elemet vagy szálat (pl.: a birtok igényel egy alanyesetű vagy datívuszos alakot) keres. A felkínálások és az igények feldolgozása során azonban túl nagy mennyiségű felkínálás keletkezhet egy adott igényhez, amelyek közül számtalan az irreleváns elem. A minőségelemző szál megfelelő specifikus jegyekkel (feature) képes szűrni a felkínálásokat, valamint ha több helyes felkínálás is van, segíthet kiválasztani a megfelelő felkínálási elemet.

A különböző szálaknak különböző minőségi jelzőkre van szüksége, valamint a felhasználót is folyamatosan tájékoztatni kell az aktuális minőségről. Ezért egy olyan minőségbecslő rendszerre van szükség, ami rugalmasan változtatható, bővíthető és ütemezhető. Továbbá a feladatoknak megfelelően a különböző jegyek különválaszthatóak legyenek. A hagyományos minőségbecslő rendszerek, mint a QuEst [12], nem tudják kielégíteni változtatás nélkül ezen követelményeket (más célt szolgálnak), ezért létrehoztunk, a hagyományos minőségbecslő rendszerekre építkezve, egy új minőségbecslő architektúrát.

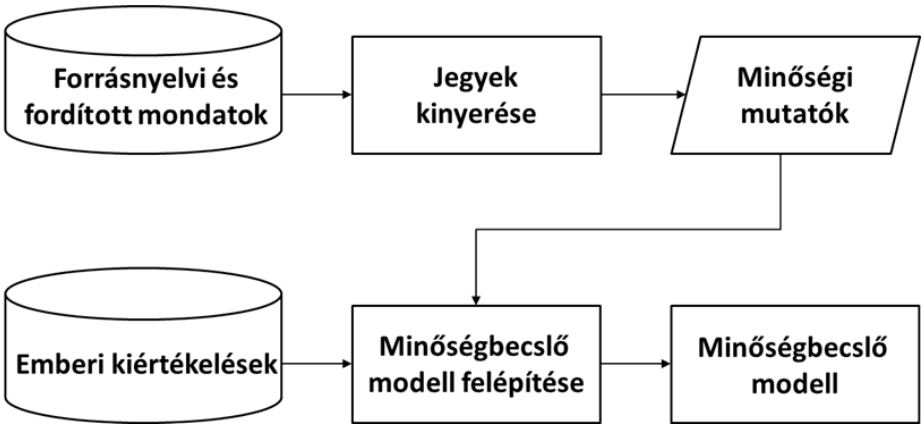
2. Kapcsolódó munkák

A hagyományos minőségbecslő módszer (lásd 1. ábra) különböző minőségi mutatókat nyer ki a forrás és a gép által lefordított mondatokból. Majd gépi tanulással betanítja a minőségi mutatókat az emberek által kiértékelt mutatókra. Az így betanított modell segítségével tudja megbecsülni az új ismeretlen mondatok minőségét. Mivel a gépi tanulás modellje emberi kiértékelésen alapszik, ezért a becsült értékek magasan korrelálnak az emberi kiértékeléssel.

Az AnaGramma egy egynyelvű elemző, ezért a minőségbecslő modellünk tanításához egynyelvű korpuszt használtunk.

A QuEst++ [11] rendszer szó szintű elemző része tartalmaz egynyelvű kiértékeléseket, többek között nyelvi modell jegyeket, szintaktikai jegyeket, célnyelvi kontextus jegyeket stb. De ezek a kiértékelések csupán egy apró részét képezik a gépi fordítást kiértékelő rendszernek és nem egy kifejezetten egynyelvű minőségbecslő rendszer.

Számtalan kutatási és üzleti ágban használatos a feladatorientált vagy szolgáltatásorientált architektúra. Ilyen területek például az elektronikus kereskedelem [7], a robotika [8], az automatikus videó megfigyelő rendszerek [5] stb. A feladatorientált architektúra előnye, hogy a modell feladatokban gondolkozik. Minden feladat egy független egység, amely önálló funkcionalitással bír. A különböző feladatok különböző erősforrásokat, valamint eszközöket használhatnak és akár párhuzamosan egyszerre többféle problémát is megoldhatnak.



1. ábra. A minőségbecslő modell

Hatékony ütemezéssel rugalmasan optimalizálhatjuk a teljesítményt és a különböző specifikus igények kiszolgálását.

A kutatásunkban a hagyományos minőségbecslő rendszert alakítottuk át a feladatorientált architektúrával.

3. πRate rendszer

A kutatásunk során implementáltunk egy egynyelvű minőségbecslő rendszert, a πRate[3] rendszert. Kettő fő modulja van a rendszernek (lásd 2. ábra): a tanuló modul és a kiértékelő modul.

A tanuló modul legfőbb feladata, hogy betanítja a minőségbecslő modellt. Ez a modul megegyezik a hagyományos minőségbecslő modell tanuló moduljával (lásd 1. ábra).

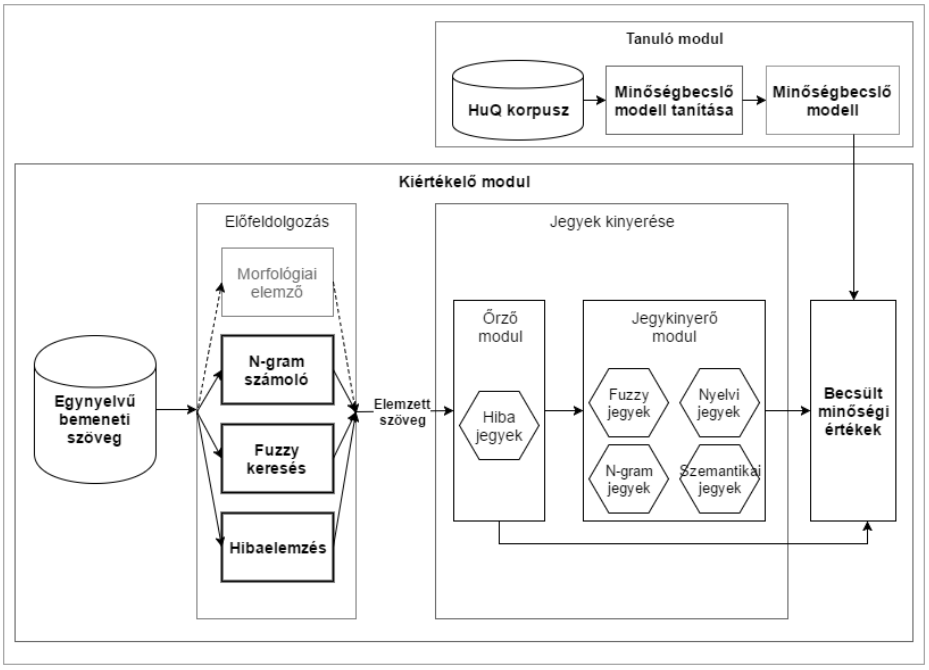
A tanításhoz emberek által kiértékelt egynyelvű korpuszt használtunk. A gépi tanuláshoz különböző nyelvi és statisztikai jegyeket használtunk: pl.: nyelvi jegyek; n-gram jegyek; fuzzy jegyek; hiba jegyek stb.

A korpuszból a jegyek segítségével kinyertük a minőségi mutatókat, majd a mutatók segítségével betanítottuk a minőségbecslő modellt az emberek által kiértékelt minőségi mutatókra.

A másik fontos modul a kiértékelő modul. Első lépésként a kiértékelő modul beolvassa a bemeneti szöveget, majd az előfeldolgozó fázisban elemezzük azt. Az így elemzett szöveget adjuk tovább a jegykinyerő (feature extraction) modulnak, amely a különböző jegyek és a betanított minőségbecslő modell segítségével előállítja a minőségi mutatókat.

A bemenet lehet nyers vagy elemzett szöveg. A feladatorientált πRate rendszer egyik előnye, hogy a morfológiai elemzés műveletét kihagyhatjuk az előfel-

[3] A kutatólaborunk a 314-es szobában található.



2. ábra. A πRate rendszer

dolgozó fázisban (a 2. ábrában a morfológiai elemző szürke), mivel az AnaGramma esetében a minőségbecslő szál már egy morfológiailag elemzett szöveget kap, amellyel így képes optimalizálni az erőforrást és ezáltal a teljesítményt.

A kiértékelő modulnak három fő része van:

– előfeldolgozó modul;
– ellenőrző modul;
– jegykinyerő modul.

A bemeneti szöveg inkrementálisan bővül. Amint a szöveg beérkezik a πRate rendszerbe, az előfeldolgozó modul elemzi a szöveget például morfológiailag elemzi (ha az még nem elemzett) a szöveget, kiszámolja az n-gram valószínűségeket stb. Majd az elemzett szöveget továbbküldi a jegykinyerő modul számára, ahol először az ellenőrző modul a hiba jegyek segítségével leellenőrzi a szöveget. Ha a szöveg hibamértéke meghalad egy megadott küszöbértéket, akkor az ellenőrző modul felhatalmazást kaphat megszakítani a folyamatot vagy szűrni, "cenzúrázni" a szöveget. Máskülönben továbbengedi a többi jegyek számára a szöveget és a saját hibaértékeit minőségi mutatóként használja fel a minőségbecslő modell.

A jegyek kinyerése után a minőségi mutatókkal a minőségbecslő modell kiszámolja a becsült értékeket (pl.: szálspecifikus értékek, aktuális mondat minősége, eddig beolvasott összes szöveg globális minősége stb.).

4. Módszerek és mérések

A minőségbecslő modell felépítéséhez egynyelvű jegyekre van szükségünk, amelyek segítségével kinyerjük a minőségi mutatókat. A tanítás során a jegyek egy egynyelvű korpuszból nyerik ki a szükséges értékeket. Majd gépi tanulással emberek által kiértékelt minőségi mutatókra tanítjuk be a modellt (lásd 1. ábra). A πRate rendszer felépítéséhez JAVA EE-t használtunk.

4.1. HuQ corpus

A minőségbecslő modell tanításához a HuQ korpuszt [14] használtuk. A HuQ korpusz 1500 magyar mondatot tartalmaz. Mind az 1500 mondatot három magyar anyanyelvű ember értékelte ki. A kiértékeléshez a Likert értékskálát használták, ebben az esetben 1-től 5-ig lehetett értékelni a mondatok minőségét. A HuQ korpusz továbbá tartalmaz még osztályzási értékeket is: BAD: 1 <= minőség <=2; MEDIUM: 2 < minőség < 4; GOOD: 4 <= minőség <=5. A korpusz vegyes témájú: film feliratok, irodalom és jog.

A kísérletünkben a HuQ korpuszt felosztottuk 500 mondatot fuzzy referenciának és 1000 mondatot minőségbecsléshez.

Az 500 mondatos fuzzy referenciakorpuszt kézzel állítottuk össze. Közel egyenlő arányban tartalmaz "BAD", "MEDIUM" és "GOOD" osztályzatú mondatokat (167 "BAD" mondat, 166 "MEDIUM" mondat, 167 "GOOD" mondat).

Az 1000 mondatot, amelyeket a minőségbecslő modellhez tettünk félre, további 90-10% arányba osztottuk fel tanító és tesztelő halmazra. A teszteléshez tízszeres keresztvalidálást használtunk.

4.2. Egynyelvű jegyek

A minőségbecslő modellünk 32 különböző típusú jegyet használ, amelyeket jellegük alapján az alábbi kategóriákba soroltuk:

- nyelvi jegyek:
  - főnevek, igék, igekötők, melléknevek, határozószók, kötőszók, névmások, névelők, indulatszók aránya a mondatban;
  - főnevek és igék aránya a mondatban;
  - főnevek és melléknevek aránya a mondatban;
  - igék és igekötők aránya a mondatban;
  - főnevek és névelők aránya a mondatban;
- n-gram jegyek:
  - a mondat nyelvmodell valószínűsége;
  - a mondat nyelvmodell perplexitása;
  - a mondat nyelvmodell perplexitása mondatvégi írásjel nélkül;
  - a mondat szótöveinek, szófaji címkéinek nyelvmodell valószínűsége;
  - a mondat szótöveinek, szófaji címkéinek nyelvmodell perplexitása;
- fuzzy jegyek:

  - A Hanna Bechara és társainak szemantikai hasonlóság kutatása [1] alapján a HuQ korpusz egyharmadát referenciakorpuszként használtuk fel. A referenciamondatok közül fuzzy kereséssel megkerestük a bemeneti szöveghez legjobban hasonlító mondatot. Majd a megtalált referenciamondathoz tartozó minőségi értékeket (Likert és osztályzási értékei) a minőségbecslő modellünkben felhasználtuk minőségi mutatóként (jegyként). A fuzzy kereséshez használtuk a Levenstein távolságot, a TER (Translation Error Rate) mértéket, a BLEU mértéket, a NIST mértéket és a szemantikai hasonlóságot mérő LSI [4] módszert és a beágyazási modelleket [10].
- hiba jegyek:
  - xml címkék aránya a mondatban;
  - nem magyar szavak aránya a mondatban;
  - ismeretlen szavak aránya a mondatban;
  - írásjelek aránya a mondatban.

Az egynyelvű jegyek és a HuQ korpusz segítségével felépítettük a minőségbecslő modelljeinket:

- LS modell: minőségbecslő modell Likert értékeket felhasználva.
- OS modell: minőségbecslő modell osztályzási értékeket felhasználva.

4.3. Mérések

A minőségbecslő modell felépítéséhez több gépi tanuló algoritmust is kipróbáltunk, amelyek közül szupport vektor regresszió adta a legjobb eredményeket, ezért a továbbiakban ezt használtuk.

Miután betanítottuk a minőségbecslő modellt, implementáltuk a πRate rendszert. A πRate rendszer előfeldolgozó fázisában: az elemzéshez használtuk a PurePos 2.0 [6] szófaji elemzőt; az n-gram számoláshoz a SRILM [13] eszközkészletet; a fuzzy kereséshez a BLEU, NIST és Levenstein mértékeket; a szemantikai hasonlóság méréséhez az LSI és a beágyazási modelleket. A kiértékelő fázisban az ellenőrző jegyek szűrték a hibás bemenetet, majd a jegykinyerő modul kiszámolta a minőségi mutatókat.

A πRate rendszer jegyhalmazára optimalizálást végeztünk el, úgy ahogyan a hagyományos minőségbecslés módszerében optimalizálni lehet a jegyek halmazát [2].

Az optimális jegyhalmaz megtalálásához a „forward selection” [15] módszert használtuk:

- OptLS halmaz: optimalizált jegyhalmaz LS modellhez.
- OptOS halmaz: optimalizált jegyhalmaz OS modellhez.

5. Eredmények és kiértékelések

A πRate rendszer kiértékeléséhez a MAE (mean absolute error - átlagos abszolút eltérés), az RMSE (root mean square error - átlagos négyzetes eltérés gyöke),

a Pearson-féle korreláció és a helyesen osztályozott egyed (Correctly Classified Instances - CCI) mértékeket használtuk.

A HuQ korpusz és a 32 jegy segítségével betanítottuk a minőségbecslő modellt és felépítettük a πRate rendszert. Az 1. táblázatban és a 2. táblázatban láthatjuk, hogy a 32 jegyhalmazzal ∼59%-os korrelációt és ∼70% helyesen osztályozott egyedet értünk el.

	Correlation	MAE	RMSE
LS modell - 32 jegy	0,5936	0,6857	0,8961
OptLS halmaz - 13 jegy	0,6278	0,6783	0,8758

1. táblázat. LS modell és OptLS halmaz kiértékelése

	CCI	MAE	RMSE
OS modell - 32 jegy	70,7%	0,2465	0,3590
OptOS halmaz - 8 jegy	71,7%	0,2544	0,3539

2. táblázat. OS modell és OptOS halmaz kiértékelése

Az optimalizálciót a „forward selection” módszerrel végeztük el. Ezt szintén láthatjuk az 1. táblázatban és a 2. táblázatban:

– Az OptLS halmaz, 13 jeggyel ∼3%-al magasabb korrelációt tudott elérni.
– Az OptOS halmaz, 8 jeggyel ∼1%-al több helyes egyedet osztályozott.

A 3. táblázatban és a 4. táblázatban láthatjuk az optimalizált jegyhalmazokat (az eredmény minőségjavulásának mértéke alapján van sorba rendezve). Általánosságban azt állapíthatjuk meg, hogy a mondatok nyelvmodell valószínűsége és perplexitása igen fontos szempont. Illetve a beágyazási modell jobban teljesített az LSI modellnél, hiszen az optimalizált halmazokba egy LSI jegy sem került bele. Láthatjuk továbbá azt is, hogy a Fuzzy egyezés modellek is előkelő helyezéseket értek el, ami azt jelenti, hogy nagyban befolyásolja az eredményt.

A 5. táblázatban láthatunk helyes és hibás becsléseket, amelyeknél fontos szempont a fuzzy egyezés. Ha a modell talál a referencia korpuszban hasonló mondatot, akkor az általa kínált értékekkel jó becslést kapunk, de ha a fuzzy kereséssel talált mondat nem hasonlít a bemenetre, akkor erősen rontja a becslés minőségét. Ezért véleményünk szerint fontos, hogy a referencia korpusz mérete nagyobb legyen a jelenleginél, valamint változatos mintákat tartalmazzon, vagyis széles skálában tartalmazzon rövid, hosszú, rossz, közepes és jó minőségű mondatokat.

Jegyek
A mondat szófaji címkéinek nyelvmodell valószínűsége
Kötőszavak aránya
Fuzzy egyezés beágyazási modellel - Likert értéke
A mondat szótöveinek nyelvmodell valószínűsége
Főnevek aránya
NIST fuzzy egyezés (beágyazási modellel) - osztályzási értéke
A mondat szófai címkéinek nyelvmodell perplexitása
Melléknevek aránya
Írásjelek aránya
Igék és igekötők aránya
Igekötők aránya
Ismeretlen szavak aránya
Fuzzy egyezés beágyazási modellel - osztályzási értéke

3. táblázat. Optimalizált 13 jegy Likert modell számára

Jegyek
Az mondat nyelvmodell valószínűsége
Az mondat nyelvmodell perplexitása
Kötőszavak aránya
TER fuzzy egyezés beágyazási modellel - osztályzási értéke
Levenstein fuzzy egyezés (beágyazási modellel - Likert értéke
Az mondat (mondatvégi írásjel nélkül) nyelvmodell perplexitása
A mondat szóteveinek nyelvmodell perplexitása
Írásjelek aránya

4. táblázat. Optimalizált 8 jegy az osztályzási modell számára

Likert értékek		Osztályzási értékek		Sentence
Emberi értékelés	Becsült érték	Emberi értékelés	Becsült érték	
4.333	4.559	GOOD	GOOD	Mahmoud eltorzította az arcát. (Mahmoud contorted his face.)
3.667	3.198	MEDIUM	MEDIUM	Megyek az öltönyt. (I am going the suit.)
2	1.683	BAD	BAD	Az elnök a magát a vége felé, a nebraskai. (The president the himself towards the end, the Nebraskan.)
2	3.531	BAD	BAD	A többi súlyos szó, és hidrokarbon létfontosságú. (The other heavy words and hydrocarbon are vital.)
5	2.520	GOOD	GOOD	Senki sem tudja. (Nobody knows.)
2	3.628	BAD	GOOD	Ők soha csinál amit. (They never does what.)

5. táblázat. Példa jó és rossz becslésre

6. Összegzés

Létrehoztuk a feladatorientált πRate minőségbecslő modellt egynyelvű természetes elemzők számára. Mivel a hagyományos minőségbecslő modell nem tudja kiszolgálni megfelelően a pszicholingvisztikai indíttatású inkrementális elemzőt, ezért a hagyományos módszert a feladatorientált architektúrára módosítottuk. Ennek előnye, hogy rugalmasan ütemezhetjük a jegyeket és hatékonyan tudjuk kiszolgálni a különböző típusú bemeneteket, igényeket és feladatokat.

A πRate rendszer tanításához a HuQ korpuszt és 32 darab jegyet használtunk. A jegyhalmazon optimalizálást végeztünk, amellyel kevesebb jeggyel tudtunk magasabb eredményt elérni. A πRate rendszerrel $\sim$60%-os korrelációt és $\sim$70% helyesen osztályozott egyedet értünk el. A πRate rendszer az AnaGramma természetes elemzőhöz készült, de más rendszerekhez is alkalmazható.

A szoftver készen áll arra, hogy az AnaGramma elemzőbe integrálják, de mivel az AnaGramma még nem készült el teljesen, a további méréseket az integrálás után tudjuk csak elvégezni.

Hivatkozások

1. Bechara, H., Escartin, C.P., Orasan, C., Specia, L.: Semantic textual similarity in quality estimation. Baltic Journal of Modern Computing, Vol. 4 (2016), No. 2 pp. 256–268 (2016)
2. Beck, D., Shah, K., Cohn, T., Specia, L.: Shef-lite: When less is more for translation quality estimation. In: Proceedings of the Workshop on Machine Translation (WMT) (2013)

3. Indig, B., Laki, L., Prószéky, G.: Mozaik nyelvmodell az anagramma elemzőhöz. In: XII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 260–270. Szegedi Tudományegyetem, Informatikai Tanszékcsoport, Szeged, Hungary (2016)
4. Langlois, D.: Loria system for the wmt15 quality estimation shared task. In: Proceedings of the Tenth Workshop on Statistical Machine Translation. pp. 323–329. Association for Computational Linguistics, Lisbon, Portugal (September 2015), http://aclweb.org/anthology/W15-3038
5. Monari, E., Voth, S., Kroschel, K.: An object- and task-oriented architecture for automated video surveillance in distributed sensor networks. In: Proceedings of the 2008 IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance. pp. 339–346. AVSS '08, IEEE Computer Society, Washington, DC, USA (2008), http://dx.doi.org/10.1109/AVSS.2008.21
6. Orosz, G., Novák, A.: Purepos 2.0: a hybrid tool for morphological disambiguation. In: RANLP'13. pp. 539–545 (2013)
7. Papazoglou, M.P., Heuvel, W.J.: Service oriented architectures: Approaches, technologies and research issues. The VLDB Journal 16(3), 389–415 (Jul 2007), http://dx.doi.org/10.1007/s00778-007-0044-3
8. Parker, L.E.: Task-oriented multi-robot learning in behavior-based systems. In: Intelligent Robots and Systems '96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on. vol. 3, pp. 1478–1487 vol.3 (Nov 1996)
9. Prószéky, G., Indig, B.: Natural parsing: a psycholinguistically motivated computational language processing model. In: 4th International Conference on the Theory and Practice of Natural Computing. Mieres, Spain (2015)
10. Siklósi, B., Novák, A.: Beágyazási modellek alkalmazása lexikai kategorizációs feladatokra. XII. Magyar Számítógépes Nyelvészeti Konferencia pp. 3–14 (2016)
11. Specia, L., Paetzold, G., Scarton, C.: Multi-level translation quality prediction with quest++. In: ACL-IJCNLP 2015 System Demonstrations. pp. 115–120. Beijing, China (2015), http://www.aclweb.org/anthology/P15-4020
12. Specia, L., Shah, K., de Souza, J.G., Cohn, T.: Quest - a translation quality estimation framework. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations. pp. 79–84. Sofia, Bulgaria (2013), http://www.aclweb.org/anthology/P13-4014
13. Stolcke, A.: Srilm - an extensible language modeling toolkit. pp. 901–904 (2002)
14. Yang, Z.G., Laki, J.L., Siklósi, B.: HuQ: An english-hungarian corpus for quality estimation. In: Proceedings of the LREC 2016 Workshop - Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem
15. Yang, Z.G., Laki, L.J., Siklósi, B.: Quality estimation for english-hungarian with optimized semantic features. In: Computational Linguistics and Intelligent Text Processing. Konya, Turkey (2016)

II. e-magyar előadások

Az e-magyar digitális nyelvfeldolgozó rendszer

Váradi Tamás[1], Simon Eszter[1], Sass Bálint[1], Gerőcs Mátyás[1], Mittelholcz Iván[1], Novák Attila[2], Indig Balázs[2], Prószéky Gábor[2,4], Farkas Richárd[3], Vincze Veronika[3]

[1] MTA Nyelvtudományi Intézet,
1068 Budapest, Benczúr u. 33., e-mail:
VEZETEKNEV.KERESZTNEV@nytud.mta.hu
[2] MTA–PPKE Magyar Nyelvtechnológiai Kutatócsoport,
1083 Budapest, Práter utca 50/a, e-mail:
VEZETEKNEV.KERESZTNEV@itk.ppke.hu
[3] Szegedi Tudományegyetem, Informatikai Intézet,
6720 Szeged, Árpád tér 2., e-mail: {rfarkas,vinczev}@inf.u-szeged.hu
[4] MorphoLogic Kft.
1122 Budapest, Ráth György u. 36., e-mail: {novak,proszeky}@morphologic.hu

Kivonat Cikkünkben átfogó ismertetést adunk az `e-magyar` rendszerről, amely a magyar nyelvtechnológiai közösség összefogásaként jött létre. Az új infrastruktúra fő célja az eddig előállított különböző eszközök továbbfejlesztése, egységesítése és egyetlen koherens technológiai láncba szervezése volt. Az interoperabilitás mellett fontos cél volt a modularitás, a nyílt rendszer és a széles körű elérhetőség. A modulok szabadon használhatók a GATE rendszer keretein belül, emellett pedig igénybe vehető az `e-magyar.hu` oldal webszolgáltatása is. Az írott szöveg feldolgozása mellett az `e-magyar` rendelkezik egy olyan résszel is, amely a beszédfeldolgozást segíti egy beszédadatbázissal és beszédelemző modulokkal.

Kulcsszavak: kutatási infrastruktúra, természetesnyelv-feldolgozás, egységesítés, GATE, integráció, interoperabilitás

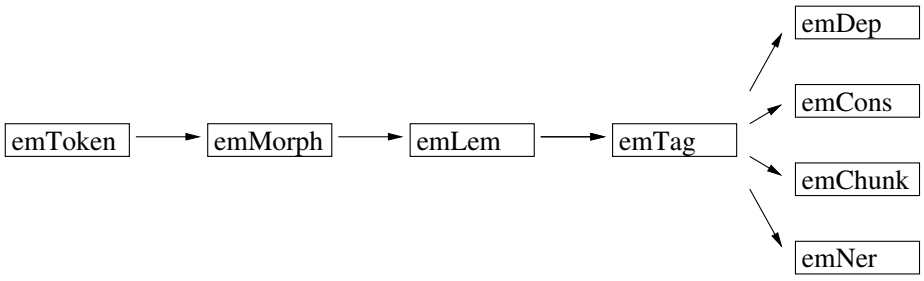
1. Bevezetés

Cikkünkben átfogó ismertetést adunk az `e-magyar` rendszerről, amely a magyar nyelvtechnológiai közösség összefogásaként jött létre 2016-ban. A munkálatok a Magyar Tudományos Akadémia támogatásával, a 2015-ben kiírt infrastruktúra-fejlesztési pályázat keretében folytak a Nyelvtudományi Intézet koordinálásával. A munkában részt vett még a Szegedi Tudományegyetem, az MTA SZTAKI, a Pázmány Péter Katolikus Egyetem, az AITIA International Zrt., valamint a MorphoLogic Kft.

Az új infrastruktúra fő célja a munkában részt vevő műhelyekben eddig előállított különböző eszközök továbbfejlesztése, egységesítése és egyetlen koherens technológiai láncba szervezése volt. Az interoperabilitás mellett fontos cél volt a modularitás, a nyílt rendszer és a széles körű elérhetőség.

A központi gondolat az `e-magyar` kialakításában az integráció volt. A magyar nyelvtechnológiai közösség külön-külön, de az utóbbi évtizedben egyre inkább

együttműködve számos kiváló erőforrást és eszközt hozott létre. Ezek a Nyelv- és Beszédtechnológiai Platform, illetve a META-NET[5] hálózaton keresztül is publikálásra kerültek. Az eszközök egy része nyílt forráskódú (ilyen a `hun*` eszközcsalád[6]), mások csak bináris formában érhetők el kutatás-fejlesztési célokra (ilyen a Humor morfológiai elemző [11]). A mostani infrastruktúra interoperábilissá tette ezeket az eszközöket abban az értelemben, hogy az infrastruktúra egyes eszközei modulárisan egymásra épülnek, vagyis önállóan is működnek, de olyan elemzési láncba is szervezhetők, amelyben zökkenőmentesen halad az adat a különböző eszközökön át. Ez azt jelenti, hogy a nyers szövegből kiindulva az `e-magyar` szövegfeldolgozó eszközlánca elvégzi a szöveg elemeinek a szegmentálását (`emToken`), megállapítja az egyes szavak tövét és teljes morfológiai elemzését (`emMorph`, `emLem` és `emTag`), majd ezek után megadja a mondatok összetevős (`emCons`), valamint függőségi elemzését (`emDep`); de ha csak egy gyors elemzésre van szükségünk, felismeri a mondatban szereplő frázisokat (`emChunk`), továbbá a szövegben előforduló tulajdonneveket (`emNer`). Az eszközök egymásra épülése az 1. ábrán látható.



1. ábra. Az `e-magyar` szövegfeldolgozó lánc elemeinek egymásra épülése.

Fontos megemlíteni, hogy a magyarra már létezik egy szövegfeldolgozó eszközlánc, a `magyarlánc` [25], amely szintén megvalósítja ezt a moduláris architektúrát, de egy zárt rendszeren belül. Az `e-magyar.hu` fejlesztésénél fontos szempont volt, hogy nyílt rendszer legyen, vagyis hogy az infrastruktúra egésze és annak minden eszköze külön-külön is elérhető, letölthető, világos licenccel publikált és kutatás-fejlesztési célra, de adott esetben üzleti felhasználásra is ingyenesen használható legyen.

A láncban részt vevő szoftverek licence GNU GPLv3 vagy GNU LGPLv3, a nem-szoftver elemekre vonatkozó licenc pedig Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA), kivéve az `emMorph` morfológiai elemző alatt működő adatbázist, amelyre az üzleti felhasználást kizáró Creative Commons Attribution-NonCommercial-ShareAlike 4.0 (CC BY-NC-SA) licenc vonatkozik.

[5] `http://www.meta-net.eu/`
[6] `http://hlt.bme.hu/en/resources/hun-toolchain`

Az e-magyar nem csak a nyelvtechnológiai szakma vagy a nyelvtechnológiát használó ipari fejlesztők igényeit kívánja szolgálni. A szakmabeli felhasználók mellett támogatjuk a számítógépes eljárások iránt fogékony, de a nyelvtechnológiában nem jártas diákok és kutatók körét is: a bölcsészet- és társadalomtudomány művelőit, valamint akár az érdeklődő nagyközönséget is. Nekik két szinten kínálunk támogatást. Egyrészt egyedi igények alapján különböző elemzőláncokat állíthatnak össze az e-magyar eszközeinek a felhasználásával a GATE szövegelemző rendszer keretein belül. Másrészt igénybe vehetik az e-magyar.hu oldal webszolgáltatását, amely rövidebb szövegek azonnali elemzését végzi.

Az infrastruktúra két fő részből áll. Az írott szöveg feldolgozása mellett az e-magyar rendelkezik egy olyan résszel, amely a beszédfeldolgozást segíti egy beszédadatbázissal és beszédelemző modulokkal. Ennek megfelelően alakul a cikk felépítése is. A szövegfeldolgozó rész moduljainak ismertetése a 2. fejezetben található, míg a 3. fejezet a beszédarchívumot és a beszédfeldolgozó eszközöket mutatja be. A 4. fejezetben ismertetjük az infrastruktúra integrálását a GATE keretrendszerbe, az 5. fejezet pedig a webes felületet írja le.

2. A szövegfeldolgozó modulok

Az e-magyar.hu a szöveg automatikus feldolgozása során a szöveget először tokenekre bontja, és megállapítja a mondatok határát; bővebben lásd a 2.1. fejezetben. Ezután megkapjuk az egyes szavakról a morfológiai információkat; ezt a lépést ismerteti a 2.2. fejezet. Mivel a magyar szóalakok jelentős részének több lehetséges elemzése van, szükség van egy egyértelműsítési lépésre; erről szól a 2.3. fejezet. Az egyes mondatok mondattani elemzése kétféleképpen is megtörténik; ezt a két modult mutatja be a 2.4. fejezet. Egy következő lépésben a főnévi csoportokat is azonosítja egy erre a célra készített modul, lásd részletesebben a 2.5. fejezetben. Végül a lánc utolsó tagja megjelöli a tulajdonneveket, ami a 2.6. fejezetben kerül bemutatásra.

2.1. Tokenizáló és mondatra bontó

Az emToken tokenizáló felépítésében és szabályaiban közel áll a magyar nyelvű szövegek tokenizálására széles körben használt Huntoken-hez[7]. Ahhoz hasonlóan az elemzendő szöveg több elemzőrétegen megy át, amelyek elvégzik a mondatszegmentálás, majd a tokenizálás feladatát. Az egyes elemzőrétegek mögött a Quex[8] lexergenerátor dolgozik.

A tokenizáló bemenetéül UTF-8 kódolású sima szöveg szolgál. A program kezeli az összes olyan karaktert, amely a latin, a görög és a cirill karaktereket, valamint a szimbólumokat tartalmazó Unicode-kódtáblákban szerepel. A kimenet szintén UTF-8 kódolású szöveg. Kimeneti formátumként az XML és a JSON közül lehet választani (a kimenet önmagában nem valid, csak csonk, gyökér elemet és

[7] http://mokk.bme.hu/resources/huntoken/
[8] http://quex.sourceforge.net/

fejlécet nem tartalmaz). A használt címkekészlet a `Huntoken`-t veszi alapul. Az **s** jelöli a mondatokat, a **w** a szavakat, és a **c** a punktuációk címkéje. Egy új címke is bevezetésre került: a **ws** a szóközjellegű karaktersorozatokat jelöli. Az `emToken` ugyanis kimenetében megőrzi a szóközöket is, ez teszi lehetővé a feldolgozott szöveg detokenizálhatóságát. Az `emToken` részletes ismertetéséhez lásd még: [10].

2.2. Morfológiai elemző és lemmatizáló

A rendszer részét képező új magyar morfológiai elemző (`emMorph`) implementációja a véges állapotú transzducertechnológiát alkalmazó `HFST` rendszer [9] felhasználásával valósult meg. A morfológiai elemző adatbázisa elsősorban az eredetileg a Humor morfológiai elemző motorhoz [16] készült magyar morfológiai adatbázison alapul [11], amelyet kiegészítettünk a `morphdb.hu` adatbázisban [23] szereplő szavakkal. A morfológiai leírást kezelő keretrendszer egy procedurális szabályrendszer felhasználásával magas szintű és redundanciamentes morfémaleírásokból állítja elő az egyes morfémák lehetséges allomorfjait, azok jegyeit és azokat a jegyalapú megszorításokat, amelyeknek az egymással szomszédos morfok között teljesülnie kell. Emellett a helyes szószerkezetek leírását egy kiterjesztett véges állapotú szónyelvtan-automata ábrázolja.

Az eredeti Humor ezeket az allomorflexikonokat, az allomorfok közötti szomszédossági megszorításokat és a véges állapotú szónyelvtan-automatát közvetlenül használja a szóalakok elemzése közben. Az új `HFST`-alapú implementációban mindezek az adatszerkezetek egyetlen véges állapotú transzducerben jelennek meg [12]. Az eredeti Humor-formalizmus szónyelvtan-automatáját a véges állapotú leírásban a *flag diacritcs* konstrukció alkalmazásával ábrázoltuk. Ez a leírás tartalmazza a morfémák közötti nem lokális megszorításokat is.

A nagybetűsített és a csupa nagybetűvel írt szóalakok kezelésére a Humorban külön mechanizmus szolgál: a lexikonban kisbetűsítve tároljuk a morfok felszíni alakjait, és a nagybetűsítést bitvektorok írják le. A nagybetűsíthető morfémákat a szónyelvtan-automatában levő leírás adja meg. A `HFST`-ben ezzel szemben ezeket a transzformációkat is újraíró szabályokkal lehet megadni. Ezeknek a szabályoknak a lexikonnal való kompozíciója a lexikon méretének és ezzel az elemző egyébként is elég jelentős futásidejű memóriaigényének megháromszorozódását eredményezi. Ezért kezdeményeztük a `HFST` elemzőmotorjának fejlesztőjénél, hogy implementáljon egy olyan mechanizmust, amely lehetővé teszi transzducerek futásidőbeli dinamikus kompozícióját úgy, hogy az egyik (itt a kisbetű–nagybetű-konverziót végző) transzducer kimenetét a hozzá kapcsolt következő transzducer (itt a morfológiai elemző) bemenetére vezeti. A módszer elenyésző elemzésisebesség-csökkenés árán harmadára csökkenti az elemző memóriaigényét.

A morfológia az összetett és képzett szavak esetében az összetételi tagokat, illetve a képzőket is azonosítja. Amennyiben az összetett vagy képzett szó a lexikonba egyben is fel van véve, több elemzés is kijöhet, amelyek különböző részletességű elemzését adják az adott szónak. A morfológiai elemzésre épülő és a nyelvi elemzés egyéb szintjeit végző eszközöknek általában nincs szükségük

ilyen részletes elemzésre, hanem csak az adott szó lemmájára és morfoszintaktikai jegyeire. A lemma magában foglalja a szóban levő töveket és képzőket is.

A lemma előállításához szükség van a tövet alkotó morfok lexikai és felszíni alakjára is. A `HFST` rendszer morfológiai elemzést végző eszközei (a `hfst-lookup`, illetve a `hfst-optimized-lookup`) alapesetben a felszíni alakot (illetve annak szegmentálását) nem adják vissza, így a képzőt tartalmazó tövek alakja nem mindig számítható ki. A `hfst-lookup` fejlesztője kérésünkre kiegészítette az eszközt egy olyan funkcióval, amely az elemzett szót alkotó morfémák felszíni és mögöttes alakját egyszerre adja vissza. Ez lehetővé tette, hogy létrehozzuk a morfológiai elemző kimenetére épülő `Java` nyelven implementált, ezért platformfüggetlen lemmatizáló eszközt (`emLem`), amely a tőalkotó elemek összevonásával kiszámolja az adott elemzéshez tartozó lemmát, annak eredő szófaját, és ehhez hozzáadja a nem tőalkotó morfémák által hordozott morfoszintaktikai jegyek címkéit. Az azonos lemmát, szófajt és egyéb címkesorozatot eredményező különböző részletességű elemzések a lemmatizáló kimenetén egyetlen elemzésként jelenhetnek meg, ezek a magasabb nyelvi szinteket feldolgozó elemzők számára ekvivalensek. Ugyanakkor a lemmatizáló képes a részletes elemzések visszaadására is úgy, hogy az az elemzést alkotó morfok felszíni alakját is tartalmazza olvasható formában. A lemmatizáló viszonylag bonyolult algoritmust valósít meg, amely nem triviális morfológiai konstrukciók (pl. ikerszavak) és különleges beállítások (pl. ha az igenévképzőket nem tekintjük tőalkotónak) esetén is helyes lemmát ad.

A korábbi magyar morfológiai elemzők általában az adott elemzőhöz *ad hoc* módon kifejlesztett címkekészletet használtak. Az `emMorph` morfológiai elemző és az `emLem` lemmatizáló kimenetén megjelenő címkekészletet ezzel szemben a nyelvészeti leírásokban elterjedten használt lipcsei notációnak megfelelő készlethez igazítottuk. A címkék meghatározásakor a Leipzig Glossing Rules-ra [3] és az ott leírtakat kiegészítő lényegesen kibővített listára[9] támaszkodtunk, amelyet kiegészítettünk a hiányzó (elsősorban képzőkkel kapcsolatos) címkékkel. Az elemző és a lemmatizáló által generált annotációval kapcsolatban lásd még: [13].

2.3. Morfológiai egyértelműsítő

A morfológiai egyértelműsítés két fő komponensből áll. Az első komponens a morfológiai címke, míg a második a címkéhez tartozó lemma meghatározása. Az előbbit a Thorsten Brants által a `TnT`-ben [2] bevezetett HMM-alapú módszer szolgáltatja, amelyet nyílt forráskódú, tanulmányozható és továbbfejleszthető implementációban `HunPos` néven ismert meg a világ [7]. A `HunPos` hátránya, hogy csak a morfológiai címkét határozza meg, a lemmát nem. Ezért volt szükség a rendszer újraimplementálására `PurePos` néven [14], amely egy lineáris modell segítségével képes kombinálni a lehetséges szótöveket és a megtalált címkéket. Ezt a programot technikai újításokkal és sebességbeli optimalizálással fejlesztettük tovább `emTag` néven, hogy egy kiforrott, robusztus rendszert hozzunk létre.

A legnagyobb problémát a tanítóanyagban lévő szóalakok, illetve azok hiánya okozza. Mivel a tanítóanyag nem tartalmaz minden lehetséges alakot, szükség

[9] `https://en.wikipedia.org/wiki/List_of_glossing_abbreviations`

van egy morfológiai elemzőre, amely a statisztikai címke-guesser segítségére siet, és a képzési szabályok segítségével kiszűri a lehetetlen alakokat.

A bemeneti szavakat az `emTag` három osztályra osztja. A tanítóanyag segítségével az adott szóhoz legenerálja a címkevalószínűségeket, amennyiben a címke előfordult a tanítóanyagban. Ezt követően a kapott elemzéseket elmetszi a morfológia által adott elemzésekkel, hogy a címkézés hamis ágait lenyesse. Amennyiben az eredmény egyelemű halmaz, az egyértelműsítés megtörtént, más esetben a Viterbi-algoritmusra bízzuk az egyértelműsítést. Ha az adott szó nem szerepelt a tanítóanyagban vagy a morfológiában, vagy üres halmaz a morfológiával közös metszete, akkor a statisztikából tanult ragozási szabályok alapján a legvalószínűbb elemzéseket rendeli a szóhoz, akár az átmenetvalószínűség romlása árán is. Ez a működés azt feltételezi, hogy a morfológia és a tanítóanyag címkézési szisztémája szinkronban van. Ha előfordul olyan eset, hogy a morfológia által ismert, de a tanítóanyagban nem szereplő változat lenne a helyes, akkor további bonyolult simítási modelleket kellene alkalmazni, amely nem várt mellékhatásokkal járhatna. Az ilyen esetekben egyszerűbb hozzáadni néhány mondatot a tanítóanyaghoz, hogy a két forrás szinkronban legyen.

2.4. Összetevős és függőségi szintaktikai elemző

A mondatok összetevős szerkezeti elemzése azt tárja fel, hogy a szavak egymással kombinálódva milyen kifejezéseket alkotnak, illetve hogyan állnak össze egy mondattá; a függőségi elemzés pedig a mondatok szerkezeti egységei közötti függőségi viszonyokat (pl. alany, tárgy, jelző) tárja fel. A szintaktikai elemzők a már tokenizált és morfológiailag egyértelműsített mondatokat kapják meg bemenetként, felhasználva a korábbi modulok kimenetét.

A morfológiailag gazdag nyelvek, köztük a magyar, szintaktikai elemzésére hozták létre a Statistical Parsing of Morphologically Rich Languages (SPMRL) workshopsorozatot. Az elemzőláncba beépített rendszer a workshop keretében megrendezett SPMRL 2014 Shared Task első helyezést elért rendszere által bemutatott technikákra épül. A rendszerbe a valószínűségi környezetfüggetlen nyelvtanokat alkalmazó Berkeley Parser [15] egy módosított változatát [20] integráltuk. A magyar nyelv gazdag morfológiájának köszönhetően a szóalakok száma rendkívül nagy, ezért a tanítóhalmazban nem vagy csak ritkán látott szóalakokat lecseréljük a szófaji egyértelműsítés során megkapott fő szófaji kódra. A Berkeley Parser tanítása során kis mértékben szerepe van a véletlennek is, ennek a véletlennek a kiküszöbölésére 8 különböző modellt tanítottunk (eltérő random seed mellett), és predikáláskor a különböző modellek által egy mondatra adott valószínűségek szorzatát vettük, így kiátlagolva a véletlen szerepét.

A `magyarlanc` [25] moduljai között szerepel egy függőségi elemző is, mely a Bohnet parser [1] nevű nyelvfüggetlen függőségi elemzőre épül, a Szeged Dependencia Treebanken [24] betanítva. Ezt az elemzőt integráltuk az `e-magyar` rendszerébe. Ugyanakkor mindkét szintaktikai elemző esetében szükségesnek bizonyultak kisebb átalakítások, hogy azok az `emMorph` modul által kiadott elemzéseket hasznosítani tudják.

A Szeged Treebank eredetileg az MSD kódrendszert alkalmazza, illetve elkészült a Universal Dependencies keretrendszerben alkalmazott morfológiai annotációra való konvertálása is. Az `emMorph` ugyanakkor egy ezektől különböző formátumot használ, és több nyelvészeti elvben is eltér a fenti kódrendszerektől, illetve a Szeged Korpusz annotációs elveitől. Ezért annak érdekében, hogy továbbra is a Szeged Korpuszt tudjuk tanítóadatként használni, szükségesnek bizonyult új konverterek létrehozása. Így összevetettük az `emMorph` nyelvészeti elveit az MSD 2.5 kódrendszer, illetve a UD alapelveivel. Az összehasonlítás eredményeképpen a Szeged Korpusz morfológiai annotációját átalakítottuk az `emMorph` formátumára. Ugyanakkor az elemzőláncban szereplő összetevős és függőségi elemzők UD vagy MSD 2.5 formátumú morfológiai kódokat várnak inputként, így ezen elemzési lépésekhez az `emMorph` elemzését automatikusan visszaalakítjuk UD és MSD 2.5 formátumra.

2.5. Sekély szintaktikai elemző

Az `emChunk` modul sekély szintaktikai elemzést valósít meg, vagyis azonosítja a mondatot alkotó frázisokat, de a köztük levő viszonyokról, illetve az általuk a mondatban betöltött funkciókról nem mond semmit. Ez utóbbit megteszi a függőségi elemző, amelyet a 2.4. fejezet ismertet.

Az `emChunk` modullal a mondatban található maximális főnévi csoportokat (NP-ket) nyerjük ki, vagyis olyan NP-ket, melyek nem részei egy magasabb szintű NP-nek sem. Tervbe van véve egy másik működési mód integrálása is, amelynek során minden típusú frázist azonosítunk a mondatban, így a főnévi csoportok mellett például a határozói (AdvP) és a névutói csoportokat (PP) is.

Az `emChunk` modul a `HunTag3`-ra [6] épül, amely egy maximum entrópiát és HMM-et használó, többféle szekvenciális címkézési feladatra alkalmas rendszer. Elődje a `HunTag`[10], amely többek között magyar nyelvű tulajdonnév-felismerésre [19] és sekély szintaktikai elemzésre [17] volt használva. Felhasználási köre igazából csak a tanítóadaton múlik. A gold standard chunk címkékkel ellátott tanítóadat a Szeged Treebank [4] összetevős elemzéséből lett a megfelelő formátumra konvertálva.

2.6. Tulajdonnév-felismerő

Az `emNer` automatikus tulajdonnév-felismerő rendszer azonosítja a folyó szövegben található tulajdonneveket, és besorolja őket az előre meghatározott névkategóriák valamelyikébe (személynév, intézménynév, földrajzi név, egyéb). Az elemzőlánc többi lépéséhez hasonlóan, az előző szinteken már feldolgozott szövegekkel dolgozik, vagyis a bemeneti szöveg tokenekre és mondatokra van bontva, valamint minden egyes tokenhez hozzá van rendelve a töve és a teljes morfológiai elemzése. Ezek az információk szükségesek a tulajdonnév-felismerő rendszer hatékony működéséhez [19].

[10] `https://github.com/recski/HunTag`

Az `emChunk`-hoz hasonlóan, e mögött a modul mögött is a `HunTag3` szekvenciális címkéző rendszer fut. A gold standard tulajdonnévi címkékkel ellátott tanítóadat a Szeged NER korpusz [21], illetve annak az új morfológiai formalizmusra konvertált változata volt. A nyelvmodell a teljes korpusz felhasználásával készült.

3. Beszédfeldolgozás

Az `e-magyar` beszédtechnológiai része tartalmaz egy beszédarchívumot, valamint három, az automatikus beszédfeldolgozást támogató modult.

Az `e-magyar` Nyílt Beszédarchívum (Open Speech Archive, `emOSA`) létrehozásával három fő célunk volt. Az első és legfontosabb a magyar beszédtechnológiára annak kezdetei óta jellemző zárt kutatási és publikációs modell felváltása egy szabad, nyílt forrású modellel. Második célunk a hagyományos, gondosan felcímkézett, és mind artikulációsan, mind aukusztikailag tiszta adatokon alapuló felügyelt tanulási módszerek felváltása gyengén felügyelt, illetve felügyeletlen módszerekkel. Harmadik célunk pedig egy a digitális bölcsészeti munkát, elsősorban a szociológiát, történelemtudományt és etnográfiát beszédtechnológiai oldalról támogató platform alapjainak megteremtése.

Az `emSad` beszéddetektáló modul beszédszegmentálást végez audió fájlokon. A fájlokat háromféle szegmensre bontja: beszéd, csend és zaj. Ez az első lépés minden további beszédfeldolgozási művelet előtt. A következő modul a beszélődiarizáló (`emDia`), amely egy több beszélő beszédét tartalmazó hangfelvétel esetében arra a kérdésre ad választ, hogy ki mikor beszélt. Képes tehát különbséget tenni a beszédhangok között, és felismerni, amikor az egyik beszélő átveszi a szót a másiktól. A harmadik modul, az `emPros` (korábbi nevén ProsoTool [22]) program az élőnyelvi kommunikációban előforduló verbális megnyilatkozások intonációjának elemzésére és lejegyzésére szolgál. Az archivált hangfelvételekből kinyerhető akusztikai paraméterek beszélőnkénti feldolgozása és stilizálása után az interakcióban részt vevők egyedi hangterjedelméhez viszonyítva címkézi fel a megnyilatkozások hanglejtését. Elsősorban több résztvevős interakciók elemzéséhez készült, de felolvasott szövegek, monologikus közlések feldolgozására is használható. Az `e-magyar` teljes beszédtechnológiai részének részletesebb leírásához lásd még: [8].

4. GATE-integráció

Az `e-magyar` szövegfeldolgozó láncát alkotó, 2. részben ismertetett eszközöknek egy egységes elemzőláncba való integrálását a GATE keretrendszerben [5] valósítottuk meg. Az integráció során az alapvető feladat az volt, hogy a modulokat alkalmassá tegyük arra, hogy a bemenetüket a GATE *offset*-alapú annotációs modelljének megfelelő formából vegyék, és a kimenetüket is ebben a formában állítsák elő. Ehhez minden eszközhöz GATE-es wrappert készítettünk, amely elvégzi a szükséges adatkonverziókat. Szükséges volt ezen túl a nem `Java` nyelven

írt eszközök illesztése a `Java` nyelvű keretrendszerhez: ezt a más nyelvű progamok binárisának közvetlen hívásával oldottuk meg.

Az eszközök az 1. ábra szerint épülnek egymásra: a tokenizálóból, morfológiai elemzőből, lemmatizálóból és egyértelműsítőből álló alapvető kötött feldolgozólánc eredményére épülnek a további eszközök, melyek már egymástól függetlenül futtathatók.

Az eszközöket további kényelmi eszközök egészítik ki. Az egyik a részletes morfológiai elemzésből állít elő egy ember számára is olvasható formát. A másik a függőségi elemzés alapján külön megjelöli az elváló igekötőt, és megadja az igekötős igei tövet. A harmadik pedig az `emChunk` és az `emNer` eszközök által szolgáltatott `IOB`-típusú (konkrétan `BIE-1`) kódolást alakítja kényelmesebben kezelhető önálló (`NP` és `NE`) annotációkká.

Az eszközlánc négyféleképpen használható. A honlapon keresztül (lásd az 5. fejezetet) egy rövidebb szöveget egyszerűen bemásolva kipróbálhatjuk az eszközláncot. Szövegelemzéshez, digitális bölcsészeti kutatáshoz a GATE rendszer *GATE Developer* nevű grafikus felhasználói felületét ajánljuk, amelybe az `e-magyar` lánc egyszerűen telepíthető. A telepítés leírása elérhető a `https://github.com/dlt-rilmta/hunlp-GATE` oldalon a teljes rendszerrel együtt. Itt lehetőség van a rendszer továbbfejlesztésére, vagyis az elemzőlánchoz saját készítésű modulok is hozzáadhatók. Nagyobb korpuszok feldolgozásához a GATE parancssori hozzáférését ajánljuk, ennek használata szintén az említett honlapon található, szükséges hozzá a `github` repozitórium használata. Negyedik módszerként használatba vehető az ún. *gate-server* is, ez szintén parancssori technológia, és ez az egyébként, amely a honlap mögött is üzemel. Az integrációról és a rendszer használatáról részletesebben lásd még: [18].

5. Webes felület

A projekt célkitűzései között szerepelt, hogy az elemzőlánc hozzáférhető és érdemben használható legyen olyan felhasználók körében is, akik nem feltétlenül járatosak az informatika területén. Ennek az igénynek igyekszik megfelelni az `e-magyar.hu` webes szövegelemző szolgáltatása[11], amely lehetővé teszi, hogy egy webes interfészen keresztül bárki egyszerűen kipróbálhassa az egyes elemző modulokat vagy akár a teljes elemzőláncot, anélkül, hogy ehhez a böngészőn kívül bármilyen egyéb szoftvert használnia kellene.

A szövegelemző egy olyan webszolgáltatásra épül, amely a GATE-es könyvtárakat használja, bemenetként az elemzett szöveget és a futtatni kívánt elemző modulok listáját várja, kimenetként pedig a GATE által generált, az annotációkat tartalmazó `XML`-t adja vissza. A weboldal a visszakapott `XML`-t feldolgozza, és a kinyert adatokat megjeleníti egy könnyen értelmezhető, vizualizált formában.

Az elemző felülete két fő részből áll: egy bemeneti és egy kimeneti panelből. A bemeneti panelen található szövegmező segítségével tudja megadni a felhasználó az elemezni kívánt szöveget (a bevihető szöveg hossza jelenleg 6000 karakterre

[11] `http://e-magyar.hu/parser`

van korlátozva), valamint itt tudja összeállítani azoknak a moduloknak a listáját, amelyeket le szeretne futtatni a szövegen.

A feldolgozás eredménye a kimeneti panelre kerül. Az elemzés kétféle elrendezésben jeleníthető meg: 'szöveg' és 'lista' nézetben. 'Szöveg' nézetben a tokenek sorfolytonosan követik egymást, az egyes tokenekhez tartozó annotációk kis buborékokban jelennek meg a kurzort egy adott token fölé mozgatva vagy kattintásra. Elváló igekötők esetében az igei tagot tartalmazó token is automatikusan kiemelődik. 'Lista' nézetben az egyes tokenek táblázatos formában, külön sorban egymás alá, az elemző modulok által hozzáadott annotációk pedig az egymást követő oszlopokba kerülnek. Ez utóbbi elrendezés alkalmasabb sok információ együttes megjelenítésére. Ebben a nézetben lehetőség van a tokenek szűrésére különböző szempontok alapján: a felhasználó szűrhet szóalakra, a morfológiai elemzés egy részletére (az `emMorph` kimenete), szófaji címkére (az `emTag` kimenete) és grammatikai funkcióra (az `emDep` kimenete). Mindkét nézetben lehetőség van az egyes modulok által létrehozott egy vagy több tokenből álló szegmensek kiemelésére: a tokenizáló esetében ezek a tokenek és a mondatok, a sekély szintaktikai elemző esetében a főnévi csoportok, a tulajdonnév-felismerő esetében pedig a tulajdonnevek. A szintaktikai elemzések eredménye a 'szöveg' nézetben érhető el az egyes mondatokat követő ikonokra kattintva: a függőségi elemző kimenete egy függőségi fa, az összetevős elemező kimenete pedig egy ágrajz formájában.

Az elemzés eredményét a felhasználó letöltheti magának további felhasználásra. A letöltött csomag három fájlt tartalmaz: a feldolgozásra elküldött nyers szöveget sima szövegfájlként, a GATE által generált `XML`-fájlt, valamint a 'lista' nézet kivonatát `tsv` formátumban.

6. Köszönetnyilvánítás

Az `e-magyar` eszközlánc az MTA 2015. évi Infrastruktúra-fejlesztési Pályázat 2. kategóriájában elnyert támogatás segítségével valósult meg.

A munkálatokat Váradi Tamás koordinálta, a szövegfeldolgozó részt Oravecz Csaba, a beszédtechnológiai munkát Kornai András irányította. Rajtuk és a szerzőkön kívül számos kutató és fejlesztő vett részt a projektben, akik nélkül az itt ismertetett eszközlánc nem jött volna létre. Ők a következők: Ács Judit, Bobák Barbara, Both Zsolt, Falyuna Nóra, Fegyó Tibor, Kovács Réka, Kundráth Péter, Ludányi Zsófia, Makrai Márton, Miháltz Márton, Nemeskey Dávid, Pajkossy Katalin, Rebrus Péter, Schreiner József, Siklósi Borbála, Szekrényes István, Takács Dávid, Zsibrita János.

Hivatkozások

1. Bohnet, B.: Top accuracy and fast dependency parsing is not a contradiction. In: Proceedings of Coling 2010. pp. 89–97 (2010)
2. Brants, T.: Tnt: A statistical part-of-speech tagger. In: Proceedings of the Sixth Conference on Applied Natural Language Processing. pp. 224–231. Association for Computational Linguistics, Stroudsburg, PA, USA (2000)

3. Comrie, B., Haspelmath, M., Bickel, B.: The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses (2008), `https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf`
4. Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged Treebank. In: Lecture Notes in Computer Science: Text, Speech and Dialogue. pp. 123–131. Springer (2005)
5. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M.A., Saggion, H., Petrak, J., Li, Y., Peters, W.: Text Processing with GATE (Version 6) (2011), `http://tinyurl.com/gatebook`
6. Endrédy, I., Indig, B.: HunTag3: a general-purpose, modular sequential tagger – chunking phrases in English and maximal NPs and NER for Hungarian. In: 7th Language & Technology Conference, Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC '15). pp. 213–218. Poznań: Uniwersytet im. Adama Mickiewicza w Poznaniu, Poznań, Poland (November 2015)
7. Halácsy, P., Kornai, A., Oravecz, C.: Hunpos: An open source trigram tagger. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions. pp. 209–212. Association for Computational Linguistics, Stroudsburg, PA, USA (2007)
8. Kornai, A., Szekrényes, I.: Az `e-magyar` beszédarchívum. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017). p. (jelen kötetben). Szeged (2017)
9. Lindén, K., Silfverberg, M., Pirinen, T.: HFST tools for morphology – an efficient open-source package for construction of morphological analyzers. In: Mahlow, C., Piotrowski, M. (eds.) State of the Art in Computational Morphology, Communications in Computer and Information Science, vol. 41, pp. 28–47. Springer Berlin Heidelberg (2009)
10. Mittelholcz, I.: `emToken`: Unicode-képes tokenizáló magyar nyelvre. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017). p. (jelen kötetben). Szeged (2017)
11. Novák, A.: Milyen a jó Humor? In: I. Magyar Számítógépes Nyelvészeti Konferencia. pp. 138–144. SZTE, Szeged (2003)
12. Novák, A.: A Humor új Fo(r)mája. In: X. Magyar Számítógépes Nyelvészeti Konferencia. pp. 303–308. SZTE, Szeged (2014)
13. Novák, A., Rebrus, P., Ludányi, Zs.: Az `emMorph` morfológiai elemző annotációs formalizmusa. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017). p. (jelen kötetben). Szeged (2017)
14. Orosz, G.: Hybrid algorithms for parsing less-resourced languages. Ph.D. thesis, Roska Tamás Doctoral School of Sciences and Technology, Pázmány Péter Catholic University (2015)
15. Petrov, S., Barrett, L., Thibaux, R., Klein, D.: Learning accurate, compact, and interpretable tree annotation. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. pp. 433–440 (2006)
16. Prószéky, G., Kis, B.: A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In: Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. pp. 261–268. ACL '99, Association for Computational Linguistics, Stroudsburg, PA, USA (1999)
17. Recski, G., Varga, D.: A Hungarian NP Chunker. The Odd Yearbook. ELTE SEAS Undergraduate Papers in Linguistics pp. 87–93 (2009)

18. Sass, B., Miháltz, M., Kundráth, P.: Az `e-magyar` rendszer GATE környezetbe integrált magyar szövegfeldolgozó eszközlánca. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017). p. (jelen kötetben). Szeged (2017)
19. Simon, E.: Approaches to Hungarian Named Entity Recognition. Ph.D. thesis, PhD School in Cognitive Sciences, Budapest University of Technology and Economics (2013)
20. Szántó, Zs., Farkas, R.: Special techniques for constituent parsing of morphologically rich languages. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics. pp. 135–144. Association for Computational Linguistics, Gothenburg, Sweden (April 2014)
21. Szarvas, G., Farkas, R., Felföldi, L., Kocsor, A., Csirik, J.: A highly accurate Named Entity corpus for Hungarian. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06). pp. 1957–1960. ELRA (2006)
22. Szekrényes, I.: Prosotool, a method for automatic annotation of fundamental frequency. In: 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom). pp. 291–296. IEEE, New York (2015)
23. Trón, V., Halácsy, P., Rebrus, P., Rung, A., Vajda, P., Simon, E.: Morphdb.hu: Hungarian lexical database and morphological grammar. In: Proceedings of LREC 2006. pp. 1670–1673 (2006)
24. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian Dependency Treebank. In: Proceedings of LREC 2010. ELRA, Valletta, Malta (May 2010)
25. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP. pp. 763–771 (2013)

emToken: Unicode-képes tokenizáló magyar nyelvre

Mittelholcz Iván

MTA Nyelvtudományi Intézet,
1068 Budapest, Benczúr u. 33., e-mail: mittelholcz.ivan@nytud.mta.hu

Kivonat Cikkünkben az emToken tokenizáló programot mutatjuk be. Ennek főbb tulajdonságai között említhető, a széleskörű UTF-8 támogatás, a konfigurálhatóság, az automatikus tesztkörnyezet és a programkönytár által nyújtott API. Az előállított – XML vagy JSON formátumú – kimenet detokenizálható. A program forráskódja szabadon elérhető GPLv3 licenc alatt. Az emToken az `e-magyar` eszközlánc tokenizálásért felelős modulja.

Kulcsszavak: tokenizálás, mondatra bontás, természetesnyelv-feldolgozás, kutatási infrastruktúra

1. Bevezetés

Cikkünkben az `e-magyar` szövegfeldolgozó eszközlánc részeként kifejlesztett új magyar tokenizáló és mondatra bontó eszközt mutatjuk be. Az `e-magyar` rendszerről átfogó ismertetést ad [5].

Magyar nyelvű szövegek tokenizálására széles körben használják a Huntoken[1] programot. Ez lényegében unixos parancssori szűrőknek egy bash szkript által összekötött sorozata. Maguk a szűrők a Flex[2] lexergenerátorral előállított, majd lefordított bináris fájlok. Ezek a standard inputról olvassák a bemenetüket, és a standard outputra írják a kimenetüket. A bash szkript a Unixban általánosan alkalmazott pipeline mechanizmust használja ezek összekötésére. A Flex-fájlokban definiált szűrők végzik a szöveg előzetes tisztítását (pl. a HTML-karakterentitások kezelését), a mondatra bontást, a rövidítések kezelését, magát a tokenizálást és a poszttokenizálást.

A tokenizálás feladatával kapcsolatban megfogalmazható néhány fontos igény, amit a Huntoken számos erénye mellett sem tud kielégíteni. Ezek a következők:

– A Flexben nincs Unicode-támogatás, csak az egy bájtos karakterkódolásokat tudja helyesen kezelni. Ennélfogva a Flexre támaszkodó Huntoken sem képes a manapság leginkább elterjedt UTF-8-as karakterkódolású szövegek feldolgozására.

[1] http://mokk.bme.hu/resources/huntoken/
[2] http://flex.sourceforge.net/

– A Huntoken készítésének idejében a tokenizálókkal még sok olyan feladatot is elvégeztettek, ami nem tartozik szorosan véve a tokenizáláshoz, de reguláris kifejezésekkel hatékonyan megoldhatónak gondoltak. Így a Huntoken is tartalmaz olyan elemeket, amelyek ma már egyértelműen a tulajdonnév-felismerők vagy éppen a morfológiai elemzők feladata. Ide tartozik például az ISBN számok vagy a képletek felismerése, és bizonyos ragozási alakok felismerése és címkézése.
– Sok tokenizáló, köztük a Huntoken is, a szóközjellegű karaktereket egyszerűen kiszűri a szövegből, és csak a tartalmas tokeneket és punktuációkat adja vissza. Ugyanakkor a későbbi feldolgozási lépésekben hasznos lehet megőrzésük (például van olyan eset, amikor nem mindegy, hogy egy írásjel tapadt az őt megelőző szóra, vagy szóköz volt köztük). Ezt az igényt a *detokenizálhatóság* címszóval szokás jelölni: egy tokenizáló kimenete detokenizálható, ha az eredményül kapott szövegből az eredeti rekonstruálható.

Az alábbiakban bemutatásra kerülő emToken[3] a fentebb megfogalmazott igényeket kívánja teljesíteni. Ehhez támaszkodik a már meglévő Huntokenre, de jelentősen el is tér attól, ahogy azt az alábbi felsorolás mutatja.

– A Huntoken meglévő szabályai részben újra lettek implementálva, részben ki lettek hagyva (ez utóbbiak mind a tulajdonnév-felismerés vagy a morfológiai elemzés tárgykörébe sorolhatók), és készültek új szabályok is.
– Az emToken a Flex helyett lexergenerátorként a Quexet[4] használja. A Quex mellett szól, hogy kifejezetten gyors véges állapotú automatát képes generálni, és hogy támogatja az UTF-8-as kódolású bemenetek feldolgozását is, többek közt a Unicode-karakterjellemzők használatával.[5]
– Választható kimeneti formátum. Jelenleg az XML és a JSON támogatott, és tervbe van véve a TSV formátum implementálása.
– A kimeneti címkekészletbe bekerült egy, a szóközjellegű karaktereknek megfelelő címke, az új tokenizáló ugyanis – a detokenizálhatóság céljából – megőrzi a szóközjellegű karaktereket is.

2. A rendszer áttekintése

2.1. Módszer

A tokenizálás a programozási nyelvekhez készült *fordítóprogram*okban, valamint *linterek*ben és *prettyprinter*ekben is a szöveges bemenet feldolgozásának szükséges fázisa. Az erre használt szoftvert *lexikai elemzőnek* vagy röviden *lexer*nek nevezik.[6] Bár a természetes nyelvek elemzése általában eltérő elvekre épül, mint a mesterségeseké, azért a természetes nyelvek tokenizálásánál sem szokatlan a lexerek használata.

[3] https://github.com/dlt-rilmta/quntoken.
[4] http://quex.sourceforge.net/
[5] A Unicode-karakterjellemzőkről bővebben l. [6].
[6] A lexikai elemzésről bővebben l. [3, 13-24. o.].

A lexereket nem szokás kézzel megírni, általában egy erre a célra szolgáló programmal generálják őket. Ezeknek *minta–akció* párokat lehet megadni, ahol a *minta* egy reguláris kifejezés, az *akció* pedig annak a leírása, hogy mit csináljon a generált lexikai elemző, ha a minta illeszkedik. Az akciót vagy a generáló program saját kódkészletével kell megadni, vagy a generált kód nyelvén. A generáló program ezek alapján egy véges állapotú automata kódját állítja elő, amely reguláris nyelvek hatékony elemzésére képes (l. [1, 38-43. o.]).

A természetes nyelvek tokenizálását nem lehet könnyen egyetlen lexikai elemzésen belül elvégezni, ezért több lexikai elemző egymás után kötése lehet a megfelelő megoldás. Az emToken a Quex nevű lexergenerátorral készített lexikai elemzők egymás után kötött soraként gondolható el, ahol az egyik elemző kimenete a következő bemeneteként szolgál. Maga a Quex C++ nyelvű forráskódot generál, és az emToken túlnyomó részt szintén C++ nyelven készült keretrendszere az egyes elemzők összekötését és a köztük történő kommunikációt biztosítja.

2.2. Bemenet

A program bemenetéül teljesen annotálatlan, UTF-8 kódolású szöveg szolgál. A emToken nincs tekintettel a különböző (HTML vagy XML) címkékre, sem a HTML-karakterentitásként elkódolt karakterekre. Ilyen fájlok feldolgozása esetén ajánlott azokat előzőleg konvertálni sima szöveges állományokká.

A program bemenetével kapcsolatos még egy megszorítás. Megfelelő pontossággal működő tokenizáló nem készíthető el nyelvfüggetlen módon. Ennek oka, hogy a természetes nyelvek írásrendszerei jelentősen eltérnek abból a szempontból, hogy mi tekinthető bennük szó- vagy mondathatárnak. A magyar nyelv tokenizálását céljául tűző szoftvertől a nagyon eltérő írásmódok elemzése nem várható el. Az emToken a Unicode-karakterek közül csak a latin, a görög és a cirill ábécék betűit kívánja kezelni. Az ezen ábécéken kívüli betűket egy helyettesítő karakterre (U+FFFD REPLACEMENT CHARACTER,) cseréli le.

2.3. Kimenet

A program kimenete szintén UTF-8 kódolású szöveg. Jelenleg két formátum választható, az XML és a JSON.

Az emToken a következő címkekészletet használja a szöveg részeinek jelölésére:

s: Mondatok jelölése. Egy címke mindig csak egy mondathoz tartozik. (Két mondat között mindig vannak szóközjellegű karakterek.)

ws: Szóközök címkéje. A szóköz lehet mondatszintű címke is (mondatok között lévő szóközök), de lehet mondaton belüli is. A mondatokkal ellentétben a szóközcímkék bármennyi, egymás után következő szóközjellegű karaktert körülfoghatnak. Az egymást követő szóközjellegű karakterek akkor is egybe fognak tartozni, ha amúgy különböző karakterek (pl. sima szóköz és új sor karakter).

w: Szavak címkézésére szolgál. Mindig mondatokon belül található. Követheti szóköz vagy punktuáció.

c: Punktuációk jelölése. Az emToken igyekszik punktuációk bizonyos csoportjait egységként kezelni. Ez hasznos funkció a *smiley*-k esetében vagy a . . . -nál is. Más esetekben az írásjelek sorozata egyenként kerül feldolgozásra, azaz az egyes írásjelek külön-külön címkét kapnak, még ha folytatólagosan következnek is.

A emToken címkekészlete nagyban támaszkodik a Huntoken címkekészletére. Az egyetlen lényeges eltérés a `ws` címke, mivel a Huntoken sehogy nem jelöli (nem is őrzi meg) a szóközöket. Ennek az új címkének a bevezetése teszi lehetővé a detokenizálhatóságot.

2.4. API

A tokenizáló nem csak egy bináris állományként használható. A fordításkor létrejövő statikus könyvtáron és a megfelelő header állományon keresztül tetszőlegesen felhasználható más programok készítése során. Használatkor megadhatók a használandó elemzőmodulok, a bemenő szövegek forrása, a kimenet pedig vagy a standard kimenetre, vagy egy megadott `stringstream` objektumba íródik.

3. Elemzési rétegek

Az emTokenben több elemzési szint követi egymást. Az egyes rétegek közti kommunikáció céljára egy saját belső reprezentációt használtunk. Ebben az egyes címkéknek megfelelő jelek magában a szövegben, *inline* módon helyezkednek el. A hatékony feldolgozáshoz fontos volt, hogy az egyes címkéket olyan karakterekkel jelöljük, amelyek a szövegben nem fordulhatnak elő.

A Quexben írt minták esetén is a lexergenerátoroknál megszokott két elv érvényesül:

1. A leghosszabban illeszkedő mintához tartozó akció fog lefutni.
2. Két, egyforma hosszan illeszkedő minta közül a forrásban előbb szereplő fog lefutni.

A Quex-modulokban használt reguláris kifejezések értelmezésénél ezért sokszor nem elég maguknak a reguláris kifejezéseknek az értelmezése, de a többi mintát és ezek sorrendjét is érdemes figyelembe venni. A Quex nyújtotta lehetőségeket részletesen bemutatja [4].

A fejezet további részeiben az egyes Quex-modulokat ismertetjük, a feldolgozás sorrendjében.

3.1. Előfeldolgozás

Az előfeldolgozó modul feladata a bemenet ellenőrzése és szükség esetén tisztítása. Ez az érvénytelen karakterek cseréjét jelenti, illetve minden cserénél egy hibaüzenet küldését a standard hibakimenetre (*stderr*), ami tartalmazza az érvénytelen karaktert és annak helyét a bemenetben (sor- és oszlopszám).

3.2. Elválasztáskezelő

Ennek a rétegnek a használata opcionális, a `-d` parancssori kapcsoló megadásával lehet aktívvá tenni. Ekkor minden sor végi elválasztójel és az azt követő sorvége karakter törlődik, az elválasztott szavak így egyben fognak megjelenni. Ennek a modulnak a hatása detokenizálással nem fordítható vissza, illetve a modul nem biztosítja a kettős betűk megfelelő visszaállítását az elválasztás törlése után (pl. *sz-sz* $\rightarrow$*ssz*).

3.3. Mondatszegmentálás

A mondatra bontás technikailag két Quex-modulra lett felosztva. Az első felel egy nagy és összetett szabály alkalmazásával az alap mondatra bontásért, ami a legtöbb esetben elegendő. A második feladata a kivételek kezelése.

Az alap mondatra bontás során a következő szabályok érvényesülnek:

- Mondat kezdete: A kérdőjel, a felkiáltójel és a szóközjellegű karakterek kivételével bármilyen karakter állhat mondatkezdő pozícióban.
- Egy mondat tetszőlegesen sok szóközt tartalmazhat akár folytatólagosan is.
- Új sor karakterből szintén tetszőleges számút tartalmazhat, de nem folytatólagosan (két egymást követő új sor karakter után mindig új mondat kezdődik).
- Mondat vége: A mondat végén opcionálisan mondatzáró írásjel (pont, felkiáltójel, kérdőjel) lehet. A mondatzáró írásjeleknek nem kell tapadniuk, és az őket követő zárójelek és idézőjelek még az aktuális mondathoz tartoznak.
- Nincs mondathatár mondatzáró írásjelek után az alább leírt pozíciókban:
  - Ha a mondatzáró karakter után következő szó kisbetűvel kezdődik.
  - Ha a mondatzáró karakter után vessző, pontosvessző, kettőspont vagy kötőjel következik.
  - Ha a pontot közvetlenül egy szóalkotó karakter követi (például URL-ekben).
  - Ha a mondaton belüli zárójelpár a záró tagja előtt mondatzáró írásjel is van (pl. *Péter születésnapjára (idén volt a 10.) Mari nem ment el.*).

A felismert mondatokat és mondatközi szóközöket a modul felcímkézve adja tovább a mondatrabontást korrigáló modulnak.

3.4. Mondatszegmentálás-korrekció

Ez az elemzési réteg végzi a mondathatárok korrigálását. Az előző modul a fenti szabályok által megengedett helyek mindegyikére beilleszti a mondatok nyitó és záró címkéit. Mivel olyan szabályszerű eset nincs, amiben a mondatszegmentáló modul nem tesz mondathatárt oda, ahova kellene, ezért az itt megfogalmazott mintáknak és eljárásoknak csak annyi a dolga, hogy a következő helyekről töröljék a mondathatárok címkéit:

- mondatkezdő sorszámok után[7],
- dátumok közben és dátumok után,
- sorszámot követő paragrafusjel előtt,
- pont és csillag között,
- római számok után (kivéve *CD*),
- monogram után,
- pontra végződő rövidítések után.

A rövidítések azonosításához az emToken jelenleg a Huntoken eredeti rövidítéseit tartalmazó fájlnak az UTF-8-ra konvertált és kiegészített változatát használja. Tetszőleges, alternatív rövidítéslista is használható, ezt fordításkor kell a *Makefile*nak átadni. A rövidítéslistából a Quex-kódot egy python szkript hozza létre. Új rövidítéslista készítéséhez figyelembe kell venni, milyen formátumot vár a generáló szkript:

- Egész sort kommentelni a # karakterrel lehet. Sor közbeni kommentelésre nincs lehetőség.
- Minden rövidítés új sorba kell kerüljön.
- Nem kell pont a rövidítések végére (ha van, nem baj, de nem fogja használni a program).
- Minden kisbetűvel kezdődő rövidítés három változatban fog megjelenni a generált `abbrev.qx` állományban: az eredeti alakban, nagy kezdőbetűvel és csupa nagybetűvel. Ha egy rövidítés eredetileg is nagy kezdőbetűvel van megadva, akkor csak a csupa nagybetűs alakja fog pluszban létrejönni, és ha csak ez utóbbi volt eredetileg is megadva, akkor az `abbrev.qx` is csak ezt fogja tartalmazni.

Megjegyzendő, hogy a teljes feldolgozási láncban a mondatszegmentálás-korrigáló modul egymás után kétszer szerpel. Ennek az az oka, hogy az egymással átfedő szabályok közül egy „menetben" csak az egyik szabály fut le. Példának legyen két szabály: 1) ami illeszkedik az `a.b` sztringre, amiből `ab`-t csinál, és 2) ami illeszkedik a `b.c`-re, amiből `bc`-t csinál. Ekkor az `a.b.c` sztring feldolgozása során először lefut az 1)-es szabály, ami illeszkedik az `a.b`-re és az `ab.c`-t adja eredményül. Ezután az elemzés a `.c`-től folytatódik tovább, így a 2)-es szabály már nem tud illeszkedni, hiába van `b.c` részsztringje a bemenetnek. Ha másodszor is lefut az elemző, akkor az első szabály már nem fog illeszkedni, de a második igen, és előállítja a kívánt kimenetet, azaz `abc`-t. A mondatszegmentálás korrekciójakor a fenti példához hasonló szabályokról van szó, azaz egy karakterlánc egyes részláncait – a felesleges mondathatárokat – kell bizonyos pozíciókból törölni.[8]

[7] Sorszámok után már az alap mondatszegmentálást végző modul sem tesz mondathatárt, ha utána kisbetűvel vagy írásjellel folytatódik a mondat. A mondatkezdő sorszámok esetében viszont nagybetűs folytatásnál sincs mondathatár (pl. *1. Fejezet*).

[8] A Huntoken is a kétszer futtatást választotta a probléma kiküszöbölésére.

3.5. Tokenizáló

Ez a modul a bemeneti szöveg további, a megállapított mondathatárokon belüli szegmentálását végzi. A modul a felcímkézett mondatközi szóközöket változtatás nélkül adja vissza, míg a címkézetlen, mondaton belüli szóközsorozatokat felcímkézi. Hasonlóan könnyű a punktuációk kezelése is. A csoportosítható írásjelek csoportosan lesznek felcímkézve, az egyedülállók – ha nem részei egy hosszabban illeszkedő mintának – pedig egyenként.

A szavak felismeréséért felelős alapszabály a következőkre van tekintettel:

- Zárójelek kezelése. Egy zárójelpár a szó részének tekinthető, ha a pár legalább egyik tagja a szó belsejében található. Máskülönben a zárójelek különálló írásjelnek számítanak. Példák XML-annotációval: `<w>záró(jel)</w>`, `<w>(záró)jel</w>` és természetesen `<w>zár(ó)jel</w>`, de `<c>(</c><w>zárójel</w><c>)</c>`.
- Szó belsejében lehet pont, ez nem okozhat szótörést, pl. `<w>R.I.P.</w>`, de alapesetben a közvetlenül a szó előtt vagy után található pont is a szóhoz tartozik.
- Szóvégi pont nem tartozik a szóhoz, ha mondatzáró címke előtt található, kivéve, ha rövidítésről van szó. Pl. `...<w>vége</w><c>.</c></s>`, de `...<w>stb.</w></s>`
- A szóhoz kötőjellel kapcsolódó *-e* partikula mindig külön tokenként lesz címkézve, pl. `<w>van</w><w>-e</w>`.

Ezen alapszabályt egészíti ki még pár, kivételekre vonatkozó szabály. Ezek lényege, hogy bizonyos körülmények között akkor is egyben tartanak egy szót, ha az alapszabály értelmében azt több tokenre kéne bontani. A kivételek kezelésének elvi alapja hasonló a mondathatárok korrekciójához. Olyan alapszabályt kell készíteni, ami mindenhol töri a karaktersorozatot, ahol kell, de törheti ott is, ahol nem kell. Ha ez a feltétel teljesül, akkor csak olyan kivételek lesznek, ahol nem kell több sorozatra bontani a bemenetet. Az ilyen eseteket leíró minták biztos, hogy hosszabban fognak illeszkedni, mint az alapszabályban megfogalmazott minta. A lexergenerátorok – és köztük a Quex – garantálják, hogy a konkurens szabályok közül a hosszabban illeszkedő fog lefutni és ez épp az, amire a kivételek kezelésekor szükség van.

Az alábbi kivételes eseteket kezeli szavakként (vagyis látja el `w` címkével, és tartja egyben) az emToken:

- informatikai kifejezések: e-mail cím, URL, fájlnevek helyettesítő karakterrel, elérési utak;
- szavak *et* jellel (pl. *AT&T*);
- szavak aposztróffal;
- számok ponttal, vesszővel vagy spáciummal tagolva;
- tizedes törtek;
- zárójeles felsorolások, pl. *b*).

3.6. Konverterek

Az emToken két, választható kimeneti formátumát is két Quex-modul hozza létre. Ezek a tokenizáló által használt belső reprezentációt konvertálják a XML vagy JSON formátumra.

4. Tesztelés

Egy természetes nyelvi tokenizálónak nagyon sok követelményt kell egyszerre teljesítenie, sok szabályt tartamaz, melyeknek a kivételeit is helyesen kell kezelnie. Ekkora szabálytömeget nem nagyon lehet egyidejűleg fejben tartani, a tesztelés itt a szokottnál is fontosabb. Éppen ezért az emToken automatikus tesztelést biztosít a fejlesztéshez. Amikor új fordítás készül, a *Makefile* automatikusan lefuttatja az emToken teljeskörű tesztelését is. Ez azonnali visszajelzést ad a fejlesztőnek, hogy egy új szabály implementálása vagy egy régebbi módosítása milyen hatással volt a rendszerre, okoz-e hibát valahol, vagy sem.

A teszteléshez a Google Teszt C++-os keretrendszert használtuk.[9] A fordítás során a Makefile-nak megadható a teszteseteket tartalmazó fájlok halmaza, ezáltal könnyen lehet alternatív szabályokhoz alternatív teszteseteket rendelni.

5. Kiértékelés

A tokenizáló teljesítményének kiértékeléséhez a Szeged Korpusz 2.0-t [2] vettük alapul. Ami a mondatszegmentálást illeti, itt 81.648 mondatból 2.131 esetben hibázott az emToken, ami 97,39%-os pontosságot jelent. A hibák jelentős része abból ered, hogy az emToken (akárcsak a Huntoken), nem kezd új mondatot, ha a mondatzáró írásjel után kisbetűvel folytatódik a karakterlánc, szemben a Szeged Korpusszal, ahol ez gyakran előfordul.

A tokenizálásnál *word accuracy*-t mértünk, amire 99,27%-ot kaptunk (1.478.300 tokenre összesítve 10.903 hibás token jutott). Ezen hibák egy része a tokenizálási sémák különbségéből fakad (pl. az emToken a szóközökkel tagolt számokat igyekszik egyben tartani), másik része tényleges hiba.

6. Tervek

Az emToken jelenleg egy szálon fut. Ennek velejárója, hogy bármely elemzőmodul futásának feltétele, hogy az előtte lévő modulok már befejezzék a munkájukat. A tokenizáló sebességét a program többszálúsításával tervezzük továbbfejleszteni.

[9] https://github.com/google/googletest

7. Köszönetnyilvánítás

A tokenizáló elkészítésében Miháltz Márton nyújtott folyamatos szakmai segítséget. Köszönet érte. Továbbá köszönet illeti Simon Esztert a kiértékelésben és Nemeskey Dávidot a tesztelésben nyújtott segítségéért.

Az `e-magyar` eszközlánc – és benne az emToken tokenizáló – az MTA 2015. évi Infrastruktúra-fejlesztési Pályázat 2. kategóriájában elnyert támogatás segítségével valósult meg.

Hivatkozások

1. Bach, I.: Formális nyelvek. Typotex (2005)
2. Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged Treebank. In: Lecture Notes in Computer Science: Text, Speech and Dialogue. pp. 123–131. Springer (2005)
3. Csörnyei, Z.: Fordítóprogramok. Typotex (2006)
4. Schäfer, F.R.: The Quex Manual (0.65.6) (2015), elérhető: http://quex.sourceforge.net/doc/html/main.html (letöltés: 2017. január 2.).
5. Váradi, T., Simon, E., Sass, B., Gerőcs, M., Mittelholcz, I., Novák, A., Indig, B., Prószéky, G., Farkas, R., Vincze, V.: `e-magyar`: digitális nyelvfeldolgozó rendszer. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017). p. (jelen kötetben). Szeged (2017)
6. Whistler, K., Freytag, A.: The unicode character property model. Tech. rep., The Unicode Consortium (2015), elérhető: http://www.unicode.org/reports/tr23/tr23-11.html (letöltés: 2017. január 2.).

Az emMorph morfológiai elemző annotációs formalizmusa

Novák Attila[1,2], Rebrus Péter[3], Ludányi Zsófia[3]

[1] MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport,
[2] Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
1083 Budapest, Práter utca 50/a, e-mail:novak.attila@itk.ppke.hu
[3] MTA Nyelvtudományi Intézet
1068 Budapest, Benczúr utca 33
e-mail:{rebrus.peter, ludanyi.zsofia}@nytud.mta.hu

Kivonat A morfológiai elemző – lévén minden nyelvfeldolgozási lánc kezdeti lépése – a nyelvtechnológiai alkalmazásokban kiemelkedő szerepű. A kimenet értelmezése szempontjából rendkívül fontos a morfológiai elemzés kimenetének egységesítése. Cikkünkben az *emMorph* morfológiai elemzőrendszer és az *emLem* lemmatizáló implementációjának ismertetése után bemutatjuk ezen eszközök kimeneti formalizmusát, különös tekintettel a morfológiai címkékre.

1. Bevezetés

A *Nyílt, integrált magyar nyelvtechnológiai kutatási infrastruktúra fejlesztése* projekt (`e-magyar`) az MTA Nyelvtudományi Intézete vezetésével, az MTA SZTAKI, a SZTE, a PPKE és az AITIA International Zrt. közreműködésével valósult meg[4]. Célja egy olyan nyílt forrású, szabadon hozzáférhető nyelvtechnológiai infrastruktúra kiépítése volt, melynek elemei a magyar nyelv gépi elemzésének alapvető eszközeit tartalmazzák integrált, szabványos keretben [11]. A rendszer részét képezi egy új magyar morfológiai elemző, amelynek implementációja a nyílt forráskódú véges állapotú transzducertechnológiát alkalmazó hfst rendszer felhasználásával valósult meg. Jelen cikk célja a megvalósult *emMorph* morfológiai elemző[5] és az arra épülő *emLem* lemmatizáló[6] implementációjának ismertetése és az elemző, illetve a lemmatizáló kimeneti formalizmusának bemutatása, különös tekintettel a morfoszintaktikai címkékre.

2. A morfológiai elemző implementációja

A morfológiai elemző adatbázisa elsősorban az eredetileg a Humor morfológiai elemző motorhoz [8] készült magyar morfológiai adatbázison alapul [5], amelyet

[4] `http://e-magyar.hu`
[5] `https://github.com/dlt-rilmta/emMorph`
[6] `https://github.com/dlt-rilmta/hunlp-GATE/tree/master/Lang_Hungarian/resources/hfst/hfst-wrapper`

kiegészítettünk olyan szavakkal, amelyek az eredeti Humor leírásban nem, a *morphdb.hu* [10] adatbázisban viszont szerepeltek, miután az utóbbi listából kiszűrtük a hibás, illetve elhanyagolhatóan ritka szavakat. A morfológiai leírást kezelő keretrendszer egy procedurális szabályrendszer felhasználásával magas szintű és redundanciamentes morfémaleírásokból állítja elő az egyes morfémák lehetséges allomorfjait, azok tulajdonságait (jegyeit) és azokat a jegyalapú megszorításokat, amelyeknek az egymással szomszédos morfok között teljesülnie kell. Emellett a helyes szószerkezetek leírását egy kiterjesztett véges állapotú szónyelvtan-automata ábrázolja.

Az eredeti Humor elemzőprogram ezeket az allomorflexikonokat, az allomorfok közötti szomszédossági megszorításokat és a véges állapotú szónyelvtan-automatát közvetlenül használja a szóalakok elemzése közben. Az új hfst-alapú implementációban [3] mindezek az adatszerkezetek egyetlen véges állapotú transzducerben jelennek meg.

A véges állapotú transzduceren alapuló morfológiai rendszerek létrehozásánál általában az a szokásos eljárás, hogy a *lexc* lexikondefiníciós nyelv [1] segítségével létrehoznak egy alap-morfémalexikont, amelyben a morfémák valamiféle mögöttes reprezentációban szerepelnek, és a leírás e mellett tartalmaz egy az *xfst* újraírószabály-formalizmusa [1] segítségével megadott vagy a Kimmo-féle kétszintű megszorításokon alapuló szabálykomponenst, amelyet a mögöttes alakokat tartalmazó lexikonnal komponálva előáll a morfémák mögöttes és felszíni alakjai közötti, az adott kontextusban megfelelő leképezés. A hagyományos megközelítésben tehát a *lexc* lexikon és az *xfst* szabályrendszer kompozíciója hozza létre a morfológiai elemző transzducert.

Az általunk készített véges állapotú magyar morfológiai leírás ezzel szemben nem tartalmaz külön sem *xfst* újraíró szabályokat, sem Kimmo-féle kétszintű megszorításokat tartalmazó szabálykomponenst, hanem a morfémák allomorfjait és a hozzájuk tartozó szomszédossági megszorításokat folytatási osztályok formájában tartalmazó adatbázist közvetlenül egy a lexc formalizmus segítségével leírt lexikonná konvertáljuk, amely a mögöttes alakok (lemmák) és a felszíni alakok közötti helyes leképezést már tartalmazza, így további szabályokra nincs szükség. Az eredeti Humor formalizmus szónyelvtan-automatáját a véges állapotú leírásban a *flag diacritcs* konstrukció [1] alkalmazásával ábrázoltuk. Ez a leírás tartalmazza a morfémák közötti nem lokális megszorításokat is (pl. hogy a melléknevek felsőfokát jelölő prefixumot a szón belül valahol vagy egy középfokjelnek vagy valamilyen más felsőfokjelet engedélyező morfémának követnie kell). A Humor formalizmusban leírt adatbázis véges állapotú leírássá konvertálására alkalmazott algoritmusok részletes leírását l. [7] 6. fejezetében, illetve itt: [6].

3. Lemmatizálás

A morfológia az összetett és képzett szavak esetében az összetételi tagokat, illetve a képzőket is azonosítja. Amennyiben az összetett vagy képzett szó a lexikonba egyben is fel van véve, több elemzés is kijöhet, amelyek különböző részletességű elemzését adják az adott szónak. A *fejetlenség* főnév elemzésekor például

az elemző ezt egyben is megtalálja, ugyanakkor visszavezeti a *fejetlen* melléknévre, a *fej* főnévre és a *fej* igére is. Bár ezek az elemzések részben különböző szemantikai tartalmakat tükrözhetnek *(káosz, átgondolatlanság, fejnélküliség, a fejés elmaradása)*, ezek közül a jelentések közül némelyik szinte egyáltalán nem jelenik meg ténylegesen előforduló szövegekben, ráadásul a morfológiai elemzésre épülő és a nyelvi elemzés egyéb szintjeit végző eszközöknek általában nincs is szükségük ilyen részletességű elemzésre. Amire viszont szükségük van, az az adott szó lemmája (szótári töve), valamint (elsősorban a ragok, illetve bizonyos nagyon produktív képzők, pl. az igenévképzők által megtestesített) morfoszintaktikai jegyei. A lemma magában foglalja a szóban levő töveket és képzőket, mindazt, amit nem morfoszintaktikai jegyek formájában szeretnénk a további nyelvi elemzést végző eszközök számára továbbadni.

A hfst rendszer [3] morfológiai elemzést végző eszközei (a *hfst-lookup*, illetve a *hfst-optimized-lookup*) alapesetben nem olyan elemzést állítanak elő, amely közvetlenül alkalmas lenne a lemma előállítására, ugyanis kizárólag az adott elemzést alkotó morfémák mögöttes alakját és a morfoszintaktikai címkéket adják vissza, az ezeknek megfelelő felszíni alakot nem, így a képzőt tartalmazó tövek teljes szótári alakja nem mindig számítható ki. A hfst-lookup fejlesztője kérésünkre kiegészítette az eszközt egy olyan funkcióval, amely az elemzett szót alkotó morfémák felszíni és mögöttes alakját egyszerre adja vissza (illetve ténylegesen működőképessé tette ezt a korábban nem működő funkciót). Ugyan ez a kimenet emberi fogyasztásra nem igazán alkalmas[7], de lehetővé tette, hogy ennek felhasználásával létrehozzuk a morfológiai elemző kimenetére épülő Java nyelven implementált, ezért platformfüggetlen lemmatizáló eszközt (emLem), amely a tőalkotó elemek (tövek, képzők) összevonásával kiszámolja az adott elemzéshez tartozó lemmát (ehhez az utolsó tőalkotó elem kivételével a felszíni alakra van szükség), annak eredő szófaját, és ehhez hozzáadja a nem tőalkotó morfémák által hordozott morfoszintaktikai jegyek címkéit.

Az azonos lemmát, szófajt és egyéb morfoszintaktikaicímke-sorozatot eredményező különböző részletességű elemzések (pl. a *fejetlenség* főnév elemzései) a lemmatizáló kimenetén egyetlen elemzésként jelenhetnek meg, hiszen ezek a magasabb nyelvi szinteket feldolgozó elemzők számára (szófaji egyértelműsítő, szintaktikai elemző stb.) ekvivalensek. Ugyanakkor a lemmatizáló képes a részletes elemzések visszaadására is úgy, hogy az az elemzést alkotó morfok felszíni alakját is tartalmazza olvasható és jól kereshető formában[8]. A lemmatizáló viszonylag bonyolult algoritmust valósít meg, amely nem triviális morfológiai konstrukciók (pl. ikerszavak) és különleges beállítások (pl. ha az igenévképzőket nem tekintjük tőalkotónak) esetén is helyes lemmát ad.[9] Az alkalmazott lemmatizáló algoritmussal kapcsolatos további részletek [7] 4.3 fejeztében olvashatók.

[7] t:t e:e h:h e:é n:n :[/N] e:e c:c s:s k:k é:e :[_Dim:cskA/N] j:j é:e :[Poss.3Sg] t:t :[Acc]

[8] tehén[/N]=tehen+ecske[_Dim:cskA/N]=ecské+je[Poss.3Sg]=jé+t[Acc]=t

[9] Léteznek igenévképzőt tartalmazó alaktani konstrukciók, amelyekre hibás tövet kapunk, ha az igenévképző(vel azonos alakú képző)t nem tekintjük a tő részének: pl. *húsdarál(ó)*.

4. Kiértékelés

A morfológia elemző fedésével kapcsolatban Kornai András és kollégái készítettek független kiértékelést az elemző 2016 augusztusi verziójával. Bár ezen cikk célja elsősorban az elemző által generált annotáció ismertetése, itt röviden bemutatjuk ennek a kiértékelésnek az eredményét. A kiértékeléshez két nagyméretű magyar nyelvű korpuszt, az MNSZ2-t (Magyar Nemzeti Szövegtár V2.0[10]) és a WebKorpusz 2.0-t (WK2[11]) használták. A korpuszokból azokat a szavakat választották ki, amelyek legalább három MNSZ2-részkorpuszban szerepeltek, és a WebKorpuszban is legalább háromszor előfordultak. A kiválasztott 1363692 szóalak az MNSZ2 95,65%-át és a WK2 94,66%-át fedi le. A kiválasztás során a két korpusz tokenjeinek 5,12%-a esett ki. A tesztanyagból az elemző által felismert szóalakok korpusztokenekre visszavetített aránya 92,63%, a nem elemzetteké 2,25%. Kornaiék ezt az fedést „kiemelkedően jó"-nak minősítették.[12]

5. A morfológiai elemző által generált annotáció

5.1. Motiváció

A morfológiai elemzés kimenetének egységesítése rendkívül fontos a kimenet értelmezése szempontjából, legyen az elemzés automatikus vagy nyelvészeti alapú, és a kimenet feldolgozása automatizált vagy emberi erővel történő. Az ilyen kimeneti annotációs rendszerekben a morfológiai elemzők tipikusan kétfajta információt jeleníthetnek meg: morfológiait és morfoszintaktikait. A morfoszintaktikai információ megadja, hogy az adott szóalak milyen szintaktikai környezetben és funkcióban fordulhat elő, előre megadott morfoszintaktikai tulajdonságokhoz rendelt értékek használatával. A morfológiai információ megmutatja, hogy mely morfémaváltozatokból (morfokból) áll össze a szó, és ezekhez a morfokhoz mely morfoszintaktikai jegyek rendelhetők. E két információtípust tipikusan egyszerre szokták az annotációs rendszerek megjeleníteni, de különböző rendszerek különböző arányban. A két szélsőség egyikét a nyelvészeti morfo(fono)lógiai elemzés képviseli, ahol az explicite nem megjelenő morfoszintaktikai információk nem lényegesek (hiányozhatnak), viszont a morfokra való szegmentálás általában központi jelentőségű. Ezekkel szemben állnak azok a formális annotációs rendszerek, amelyekben csak morfoszintaktikai jegyek vannak, és az annotáció nem tartalmaz a morfszegmentálásra vonatkozó információt (ez utóbbira példa az ún. Universal Dependencies [4], az MSD-kódolás vagy a hunmorph rendszerben működő ún. KR-kódolás [9]). Több rendszerben a kétféle információt az annotáció egyszerre tartalmazza (pl. ilyen a Humor [5,8] vagy a Xerox magyar morfológiai elemzője), de ezek megjelenítése sokszor némileg ad hoc módon történik.

[10] `http://mnsz.nytud.hu`

[11] `http://mokk.bme.hu/en/resources/webcorpus`

[12] A jelenlegi verzió az itt ismertetettnél jobb fedést mutat, mert egy jelentős hibaosztály (Kornaiék a kötőjeles szavak egy nagy osztályára nem kaptak elemzést) megszűnt.

Ennek praktikus okai vannak: az írott szóalakok szegmentálása bizonyos esetekben szükségszerűen önkényes: pl. a *hússzal* szóalak morfokra bontásakor a *húsz* tő és a *szal* eszközhatározó-rag közötti határ meghúzása a helyesírás sajátosságai miatt sehogy sem lesz igazán jó. A Humor rendszerben használt *hússz+al* tagolás mellett praktikus (a lexikonmérettel és a jegyrendszer komplexitásával kapcsolatos) szempontok szólnak: a kétjegyű betűre végződő szavakhoz mindenképp elő kell állítani egy-egy plusz allomorfot, ugyanakkor az ezekhez kapcsolódó eszközhatározórag-allomorfból ebben az esetben elég, ha egy van a lexikonban.

Az emMorph elemző kimeneti formalizmusa kialakításakor abból indultunk ki, hogy az egyszerre kell szolgálja a számítógépes nyelvfeldolgozást és a nyelvészeti elemző munkát. Ennek megfelelően igyekeztünk arra törekedni, hogy az annotáció mind a releváns morfológiai szegmentálást, mind a szükséges morfoszintaktikai jegyeket tükrözze, és belőle ezek külön-külön is kinyerhetők legyenek. Ugyanakkor mivel az elemző alapvetően a Humor rendszer számára implementált szabályrendszeren alapszik, a szegmentálás tekintetében megmaradt néhány a Humor leírásból örökölt kompromisszum. Egy másik megszorítás az volt, hogy szerettük volna a korábban használt annotációs sémák és az új rendszer közötti konverziót lehetőleg minél teljesebb mértékben lehetővé tenni. Ezért azokat a komplex toldalékokat, amelyekhez tartozó címke a korábbi rendszerek valamelyikében nem tagolódott világosan elkülöníthető elemekre (pl. az *-i* „birtoktöbbesítő jel”-et tartalmazó birtokos végződések), nem szegmentáltuk szét különálló elemekre az új annotációs sémában sem, hanem azokat a fúziós morfémáknak megfelelő módon ábrázoltuk (l. a 5.5 részt).

Az annotációs rendszer egyben szabványosítási javaslat a magyar nyelvű automatikus morfológiai elemzők kimeneti formátumára, és a magyar alaktan nyelvészeti glosszáinak formátumára. A korábbi magyar morfológiai elemzők egyedi és mind egymástól, mind az esetleges nemzetközi szabványoktól eltérő címkéket használtak. A projekt keretében megvalósult elemző címkekészletét ezzel szemben igyekeztünk nemzetközi szabványhoz igazítani: amennyire lehetséges volt, a nyelvészeti annotációra széles körben egyfajta szabványként használt Leipzig Glossing Rules (LGR) [2] javaslatait követtük. A címkék meghatározásakor emellett az ott leírtakat kiegészítő lényegesen kibővített listára (List of glossing abbreviations = LOGA)[13] támaszkodtunk, amelyet az ezekben a dokumentumokban leírtak szellemében kiegészítettünk a hiányzó (elsősorban képzőkkel kapcsolatos) címkékkel.

5.2. Az annotáció felépítése

Míg a Leipzig Glossing Rulesban javasolt annotációs séma szerint az annotáció külön sorokban tartalmazza a morfokra szegmentált elemzett alakot és a morfokhoz tartozó morfoszintaktikai jegyeket (amely csak a tövek esetén tartalmaz alaki információt: a lemmát), a véges állapotú morfológiai elemző kimenetén ezek az elemek szekvenciálisan jelennek meg: az egyes morfok mögöttes és felszíni alakja, illetve a hozzá tartozó morfoszintaktikai címke együtt jelenik meg

[13] `https://en.wikipedia.org/wiki/List_of_glossing_abbreviations`

a kimeneten. A szegmentálás jelölésére a Leipzig Glossing Rulesban a kötőjel használatát javasolják. Ennek használata – tekintettel arra, hogy a sztenderd helyesírásban ez igen gyakran eleve a szóalak része – nem lett volna praktikus.[14] Ehelyett az elemző kimenetén szögletes zárójelbe tett morfoszintaktikai címkék jelölik implicit módon a szegmentálási határokat. A Leipzig Glossing Rulesban javasolt gyakorlattól még abban a fontos kérdésben tértünk el, hogy az LGR-t követő kiadványokban – némileg meglepő módon – gyakran egyáltalán nem használnak szófajcímkéket: a tövek szófaját semmilyen módon nem jelölik. Hogy ennek a gyakorlatnak mi az oka, azt nem érdemes találgatni, mi mindenesetre nem követtük.

Az emMorphban használt annotációban a címkék egyes alaki tulajdonságai egyértelmű összefüggésben vannak az adott morféma típusával. A tőmorfémák címkéje /-lel kezdődik (`fej[/N]` főnév), a képzőké _-sal, és a képző címkéjét követő / után a képző eredő szófaja áll (`etlen[_Abe/Adj]` névszói fosztóképző „abesszívusz"), az inflexiók címkéje pedig nem tartalmaz speciális karaktert (`t[Acc]` tárgyesetrag). A szófajcímkék elé helyezett / a morphdb.hu-ban használt KR-kódrendszerből származik, a képzők _-sal való megjelölése pedig a Humor-kódkészlet sajátossága volt.

További eltérés az LGR-hez képest, hogy az emMorph kimenete a toldalékmorfok lexikai alakjait is tartalmazza. Ez nem valamiféle absztrakt fonológiai alak, hanem azzal az allomorffal azonos, amelyet az adott toldalékmorféma akkor vesz fel, amikor a szó végén áll. Ennek elsősorban a képzők esetében van jelentősége és a lemmatizáláshoz szükséges. Az emMorphra épülő emLem lemmatizáló az adott elemzéshez tartozó lemma kiszámolásakor azt a tőalkotó morfokból állítja össze. Az utolsó tőalkotó elem a lexikai, a többi a felszíni alakjában szerepel a lemmában (1. táblázat).

surface form	**butá**	**cská**	**bb**	**já**	**tól**	**nadrág**	**ocská**	**tól**
lexical form (lemma)	*buta*	*cska*	*bb*	*ja*	*tól*	*nadrág*	*ocska*	*tól*
`abstract lex. form`	`buta`	`LVcskA`	`LAObb`	`LjA`	`LtŐl`	`nadrág`	`LVcskA`	`LtŐl`
tag	/Adj	_Dim/Adj	_Comp/Adj	Poss.3Sg	Abl	/N	_Dim/N	Abl
lemma 1	**butá**	**cská**	*bb*					
lemma 2	**butá**	*cska*				**nadrág**	*ocska*	
lemma 3	*buta*					*nadrág*		

1. táblázat. Képzett és ragozott szavak lemmatizálása

5.3. Szegmentálás és alternációk

A kötőhangzót általában az utána álló toldalékhoz kapcsoljuk:
`nap[/N]ok[Pl]at[Acc]`. Az epentetikus mássalhangzókat ezzel szemben (pl. *bőv+en, ven+ne*) általában a tőhöz számítjuk.

A morfsorozat az aktuális alakban szereplő tőallomorf részsztringjeit tartalmazza. A lemma neve viszont általában a paradigma alapalakja, mely az izoláltan

[14] Az LGR formalizmusát eleinte elsősorban a helyesírási normával nem rendelkező „bennszülött" nyelvekkel kapcsolatos terepmunkagyűjtések eredményének lejegyzésére használták.

megjelenő alakkal azonos (ha ez létezik). Váltakozó tő esetén a tőallomorf nem mindig egyezik meg a lemma nevével: pl. *fá- ∼ fa, bokr- ∼ bokor, tav- ∼ tó, nyar- ∼ nyár, ve- ∼ vesz, vol- ∼ van.* Az ikes igék esetén az alapalak (és így a lemma neve) az ikes alak, függetlenül attól, milyen tőváltozat jelenik meg a szóban forgó alakban: *laktok:* `lakik[/V]tok[Prs.NDef.2Pl]`.

Ha az alapalak is több alakban jelenhet meg (mint az sz∼d váltakozást mutató igéknél), akkor a gyakoribb alakot vesszük lemmának – az, hogy ez melyik, az egységes lemmaazonosíthatóság miatt előre rögzíteni kell minden egyes ilyen lemmánál: *növekednek:* `növekszik[/V]nek[Prs.NDef.3Pl]`.

5.4. Hiányos és helyettesítő paradigmák

Ha egy morfológiailag hiányos paradigmájú elem alapalakja hiányzik, akkor a lemma neve a morfológiailag legjelöletlenebb alak. Plurale tantum (pl. *üzelmek, bélbolyhok, légutak*) esetén ez a nem birtokos nominativusi többes számú alak. Possessivum tantum (pl. *eleje, alja, hóna, öccse*) esetén a lemma neve az egyes számú E.3 birtokos nominativusi alak. Egyes esetekben a kétféle defektivitás egyszerre érvényesül (pl. *eleik, feleink*), ekkor a lemma a többes számú E.3 birtokos alak: *eleiknek* `elei[/N]ik[Pl.Poss.3Pl]nek[Dat]`.

Az igei defektivitás azon eseteinél, ahol nem áll rendelkezésre a jelen idő kijelentő mód indefinit E.3 alak (pl. *sínyli, kétli*), akkor a definit E.3 kijelentő mód jelen idejű alak lesz a lemma neve: *sínylitek:* `sínyli[/V]itek[Prs.Def.2Pl]`.

5.5. Fúziós morfémák

Ha egy morfhoz több jegyet kell rendelni (fúziós morféma), akkor a szóban forgó jegyek egy []-en belül jelennek meg, és ponttal választjuk el őket. Például egyes birtokosjelölős alakokban a toldalék egyszerre utal a birtoklásra (Poss) és a birtok számára/személyére (pl. 1Sg): *nadrágomat* `nadrág[/N]om[Poss.1Sg]at[Acc]`. Az elemzések Humor-elemzésekre és címkékre való leképezhetősége érdekében így jártunk el néhány olyan toldalék esetében is, amelyek esetében a szegmentálás egyébként nem lenne lehetetlen (bár bizonyos dilemmák felmerülnének): (`jaim[Pl.Poss.1Sg], nátok[Cond.Def.2Pl], nátok[Cond.NDef.2Pl], tatok[Pst.NDef.2Pl], tátok[Pst.Def.2Pl]`). A zérusmorfok jelölése nem különleges, egyszerűen üres a felszíni alakjuk (és általában a lexikai is).

Az igeidőt és a módot egymással komplementáris viszonyban levőnek tekintettük, így külön kijelentő mód jegyet nem vettünk fel, hanem valamely időjegy (`Prs, Pst`) meglétéből következik a kijelentő mód.

5.6. Unáris jegyek

Vannak olyan morfoszintaktikai dimenziók, amelyeknek csak egy értéke jelenik meg – ezek az ún. unáris jegyek. Azt az információt, hogy ilyen értékkel az alak nem rendelkezik, az annotáció nem jelöli (pontosabban az adott jegy hiányával jelöli). A modális igei alakokban (pl. *adhatsz* `ad[/V]hat[_Mod/V]sz[Prs.NDef.2Sg]`)

unáris jegy áll, ahogyan az összes képzett alakban is. Ezzel szemben az inflexiós jegyek nagy része nem unáris, például az igeragozás definitsége tekintetében az `Def` jegy szemben ál az `NDef` jeggyel, az alanyesetet is megjelöljük a `Nom` jeggyel. A jelen implementációban sajátos kivételként a névszóragozás paradigmájának leírásában az egyes szám jelöletlenül maradt. Ennek oka az volt, hogy a morfokra szegmentálás szempontjából ennek a jegynek mind a tőhöz, mind a toldalékokhoz rendelése ellentmondáshoz vezetett volna.

5.7. Az alkalmazott címkék

Mint korábban említettük, az elemzőben igyekeztünk következetesen az LGR és a LOGA dokumentumokban felsorolt címkéket használni, illetve az ott megadott alternatív jelölések közül választani. Azon címkék ügyében szavazással döntöttünk, amelyekkel kapcsolatban az előkészítő fázisban nem jutottunk konszenzusra. Így született többek között az igekötők `/Prev` (preverb), a igenevek `Ptcp` a névelők `Det`, a melléknevek, illetve a számnevek `Adj`, illetve `Num` címkéje. Az alkategóriára utaló jegyek a címkén belül |-lal elválasztva jelennek meg, pl. `/Adj|Pro|Int`: melléknévi kérdő névmás (pl. *milyen*). Zárójelben szerepel a vonzatos névutók vonzatát jelölő esetrag kódja: `/Post|(Abl)`. A (szinte) azonos funkciót nem fonológiailag vagy lexikailag kondicionált módon, hanem lényegében szabadon választhatóan különböző formában kifejező toldalékok esetében a funkció mellett a formára is utal a használt címke (a formára utaló címkerész előtt mindig kettőspont áll): `EssFor:ként, EssFor:képp, EssFor:képpen`, illetve `_Adjz_Type:fajta/Adj, _Adjz_Type:forma/Adj, _Adjz_Type:féle/Adj, _Adjz_Type:szerű/Adj` (Adjz: adjectivizer 'melléknévképző'). A képzők esetében a formára sokszor egyébként is utalunk. Sőt, időnként – amikor a funkció viszonylag heterogén, illetve nem volt egyszerű egy rövid címkében egyértelműen megnevezni – csak a formára (és az eredő szófajra) utal a címke: `_Adjz:i/Adj, _Adjz:s/Adj, _Adjz:Ő/Adj, _Adjz:Ű/Adj`.

6. Konklúzió

A cikkben bemutattuk az `e-magyar` projekt keretében megvalósult új, nyílt forráskódú morfológiai elemzőeszközt. Kitértünk a lemmatizáló és a morfológiai elemző implementációjának főbb kérdéseire, majd részletesen ismertettük a nyílt forráskódú emMorph morfológiai elemző és emLem lemmatizáló kimeneti formalizmusát, az általuk generált annotációt. Az emMorph által generált annotáció formalizmusa sztenderdizált, automatikus és kézi feldolgozásra is alkalmas. A jegyek elnevezése (rövidítése) és sorrendje a nemzetközi nyelvészeti konvenciókhoz kötődik, így jól olvasható, és a nyelv ismerete nélkül is értelmezhető.

7. Köszönetnyilvánítás

Az `e-magyar` eszközlánc az MTA 2015. évi Infrastruktúra-fejlesztési Pályázat 2. kategóriájában elnyert támogatás segítségével valósult meg. Köszönetet mon-

dunk Kornai Andrásnak és kollégáinak az elemző fedésének a 4. részben ismertetett kiértékelésért.

Hivatkozások

1. Beesley, K., Karttunen, L.: Finite State Morphology. No. 1 in CSLI studies in computational linguistics: Center for the Study of Language and Information, CSLI Publications (2003)
2. Comrie, B., Haspelmath, M., Bickel, B.: The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses (2008), `https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf`
3. Lindén, K., Silfverberg, M., Pirinen, T.: HFST tools for morphology – an efficient open-source package for construction of morphological analyzers. In: Mahlow, C., Piotrowski, M. (eds.) State of the Art in Computational Morphology, Communications in Computer and Information Science, vol. 41, pp. 28–47. Springer Berlin Heidelberg (2009)
4. McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N., Lee, J.: Universal dependency annotation for multilingual parsing. In: Proceedings of ACL 2013. pp. 92–97. Association for Computational Linguistics, Sofia, Bulgaria (August 2013)
5. Novák, A.: Milyen a jó Humor? In: I. Magyar Számítógépes Nyelvészeti Konferencia. pp. 138–144. SZTE, Szeged (2003)
6. Novák, A.: A Humor új Fo(r)mája. In: X. Magyar Számítógépes Nyelvészeti Konferencia. pp. 303–308. SZTE, Szeged (2014)
7. Novák, A.: A model of computational morphology and its application to Uralic languages. Ph.D. thesis, Roska Tamás Doctoral School of Sciences and Technology Pázmány Péter Catholic University, Faculty of Information Technology and Bionics, Budapest (2015)
8. Prószéky, G., Kis, B.: A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In: Proceedings of ACL '99. pp. 261–268. Association for Computational Linguistics, Stroudsburg, PA, USA (1999)
9. Rebrus, P., Kornai, A., Varga, D.: Egy általános célú morfológiai annotáció. Általános Nyelvészeti Tanulmányok XXIV., 47–80 (2012)
10. Trón, V., Halácsy, P., Rebrus, P., Rung, A., Vajda, P., Simon, E.: Morphdb.hu: Hungarian lexical database and morphological grammar. In: Proceedings of LREC 2006. pp. 1670–1673 (2006)
11. Váradi, T., Simon, E., Novák, A., Indig, B., Farkas, R., Vincze, V., Sass, B., Gerőcs, M., Iván, M.: e-magyar.hu: digitális nyelvfeldolgozó rendszer. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017) (2017)

Az e-magyar rendszer GATE környezetbe integrált magyar szövegfeldolgozó eszközlánca

Sass Bálint, Miháltz Márton, Kundráth Péter

MTA Nyelvtudományi Intézet, e-mail: sass.balint@nytud.mta.hu, mmihaltz@gmail.com, peter.kundrath@gmail.com

Kivonat A jelen tanulmányban bemutatott új magyar nyelvfeldolgozó eszközlánc az emberi intelligenciát igénylő szövegértési feladatnak egy jelentős szeletét automatikusan valósítja meg: a szövegben rejlő információkat automatikus módon fedi fel, teszi explicitté. Egy tetszőleges magyar nyelvű szövegrészt feldolgozva megtudjuk az egyes szavak szófaját, szótövét, alaktani elemzését, a mondatok kétféle mondattani elemzését, megkapjuk a főnévi csoportokat és a tulajdonneveket. A rendszer egybegyűjti, egy egységes láncba integrálja és közzéteszi az elemzési lépéseket megvalósító számítógépes magyar szövegfeldolgozó eszközöket. Ezáltal széles körben elérhetővé, közvetlenül felhasználhatóvá válnak ezek az eszközök a különféle igényű felhasználói körök – kiemelten a humán tudományok és a digitális bölcsészet – számára. A tanulmányban áttekintjük az eszközlánc felépítését és használatát.

Kulcsszavak: szövegfeldolgozás, szövegfeldolgozó eszközlánc, szövegelemzés, GATE, integráció

1. Bevezető példa

Mit csinál egy szövegfeldolgozó eszközlánc? Mit csinál konkrétan az e-magyar digitális nyelvfeldolgozó rendszer szövegfeldolgozó eszközlánca? Magyar nyelvű írott szöveget elemez, és lát el különféle kiegészítő információkkal az elemzés eredményeképpen. Egymásra épülő eszközökből áll: az egyes eszközök működésük során felhasználják a korábbiak eredményét.

Tekintsük a következő példaszöveget:

> *Bár külföldre menekülhetett volna, nem tette meg. Támogatta a haladó eszméket, barátságban állt pl. Jókai Mórral is.*

A rendszer a szöveg automatikus feldolgozása során először megállapítja a szavak – ún. tokenek – és mondatok határát. A példában a *Támogatta* új mondatot kezd, a *Jókai* viszont nem, bár itt is pont után nagybetűs szó következik, ami tipikusan mondathatárra utal. Külön tokenként kezeli az írásjeleket, kivéve a rövidítéseknél, ahol a záró pont a rövidítés részét képezi, így a *pl.* egy egység lesz, az *is* és az azt követő pont viszont kettő.

A morfológiai elemzés megadja az egyes szavakról az alaktani információkat: a *menekülhetett* szóalak például múlt idejű ige, mely a *menekül* szótőből,

a *het* képzőből és az *ett* igeragból épül fel. A magyar szóalakok jelentős részének, akár 30%-ának több alaktani elemzése van. A rendszer a szövegkönyezet alapján automatikusan dönt ilyen esetekben, kiválasztja a helyes elemzést, ez az ún. egyértelműsítési lépés. A többértelműség sokszor nem olyan nyilvánvaló, mint a *várnak* vagy a *terem* esetében, hanem rejtetten jelenik meg: fontos, hogy példánkban a *haladó* melléknévként elemződjön, ne pedig összetett főnévként, ami valamiféle vízi élőlényekre vonatkozó járulékot jelentene.

Ezt követően megtörténik – kétféleképpen – az egyes mondatok mondattani elemzése. A függőségi elemzés eredményeként az egyes szavak egymáshoz való kapcsolatai jelennek meg, mint például, hogy a *barátságban* az *állt* igéhez kapcsolódó határozó. Az összetevős elemzés ugyanakkor a mondat egységeit adja ki: a második mondat két nagyobb egységből áll, melyek felsorolás viszonyban vannak egymással. A függőségi elemzés alapján az ige–igekötő kapcsolatok is rendelkezésre állnak, erre építve egy külön segédmodul megjelöli az elváló igekötőket, és a hozzájuk tartozó igéket, példánkban a *tette* és a *meg* kapcsolatát. A főnévi csoportokat – pl. *a haladó eszméket* – is azonosítja egy erre a célra készített modul.

Végül a lánc utolsó tagja megjelöli a tulajdonnevek fontos alosztályait, a személy-, hely- és intézményneveket, példánkban a *Jókai Mórral* nevet.

Látjuk tehát, hogy a feldolgozás során a puszta szöveg számos explicit hozzáadott információval gazdagodik.

2. A konkrét klasszikus nyelvfeldolgozó eszközök

Tekintsük át röviden a konkrét eszközöket. Az eszközökről részletesebb információt az `e-magyar.hu` honlapon, illetve az [1] tanulmányban találunk. Az eszközök elnevezése az `e-magyar`-ra utaló *em* előtagot tartalmaz. Az `e-magyar` integrált magyar szövegfeldolgozó eszközlánc jelenleg a következő eszközökből áll.

A mondatokra bontást és a tokenizálást az emToken [2] eszköz végzi. A bemenetként megadott UTF-8 kódolású magyar nyelvű szövegben megállapítja a mondat- és szóhatárokat. Eltérő módon megjelöli a szavakat és az írásjeleket. Megőrzi a szóközöket és egyéb white space karaktereket is, ezáltal lehetővé teszi a tokenizált szövegből az eredeti szöveg visszaállítását. Széles körűen fel van készítve az egyes Unicode karakterek megfelelő értelmezésére, kezelésére.

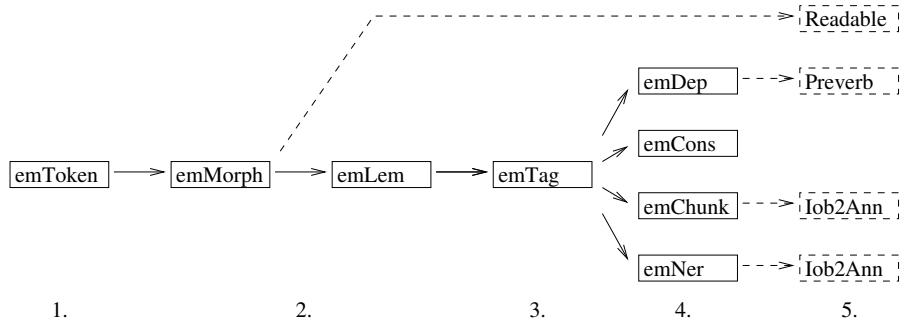
A morfológiai elemzést az emMorph [3] eszköz végzi. A szöveg minden – a tokenizáló által pontosan behatárolt – szóalakjához (tokenjéhez) hozzárendeli az összes lehetséges morfológiai, morfoszintaktikai elemzését, a szóalak aktuális környezetétől függetlenül. Megállapítja a szófaji főkategóriát, megadja a szóalak morfémákra bontásának lehetőségeit (így a szóösszetételi határokat is), és elemzést rendel az egyes morfémákhoz. Az emMorph a korábbi magyar morfológiai elemzők tudását összegzi, a leggyorsabb működést biztosító véges állapotú technológiára alapul, futtatáshoz a HFST [4] véges állapotú eszközkészletet használja.

Az eszközláncba illesztett fenti eszközök újonnan készültek, az alábbiak már korábban is meglévő eszközök legújabb verziói.

Az adott részletes morfológiai elemzéshez tartozó kívánt szótövet az elemzés alapján – a képzőket is tekintetbe véve – a morfológai elemzővel egybeépített emLem [5] szótövesítő eszköz határozza meg. A szóalakokhoz szótövet, a szótőnek megfelelő eredő szófajcímkét és inflexiós jegyeket tartalmazó egyszerűsített elemzést rendel.

A többféle lehetséges morfológiai elemzés közül a megfelelőt az emTag egyértelműsítő eszköz választja ki. Ez a *PurePOS* [6] eszköznek a *Magyarlánc* 3.0 verziójába [7] integrált változata. Ez az eszköz gépi tanulási módszerrel a szöveg minden tokenjéhez meghatározza az aktuális szövegkörnyezetben érvényes szótövét, szófaját és inflexiós jegyeit.

Az egyértelműsítőig az eszközök közvetlenül egymásra épülnek, a további eszközök viszont egymástól függetlenül alkalmazhatók. E további eszközök tokenizált és morfológiailag egyértelműsített bemenetet várnak, azaz használatuk előfeltétele az egyértelműsítőig tartó eszközlánc előzetes lefuttatása (ld. az 1. ábra folytonos vonallal írt részét).



1. ábra. Az `e-magyar` szövegfeldolgozó lánc elemeinek egymásra épülése. A fő nyelvfeldolgozó eszközök folytonos vonallal, a kiegészítő eszközök szaggatott vonallal szerepelnek.

Két különböző felfogású szintaktikai elemző kapott helyet az eszközláncban. Az emDep [7] a mondatok függőségi elemzését valósítja meg. Minden szóról megállapítja, hogy mely másik szóval áll függőségi (dependencia) viszonyban, azaz mely másik szó alárendeltje. Minden tokenhez hozzárendeli tehát a szülőcsomópontot, valamint a függőségi viszonyt leíró megfelelő szintaktikai címkét. Ezek a függőségi viszonyok az elemzett mondat szavait egy elemzési fába rendezik, az elemzési fa csomópontjai a szavak, élei pedig a függőségi viszonyok.

Az emCons [7] eszköz a mondatok összetevős szerkezeti elemzését végzi el. Az összetevős szerkezeti elemzés azt tárja fel, hogy a szavak egymással kombinálódva milyen csoportokat/kifejezéseket alkotnak, és hogy ezek a csoportok/kifejezések hogyan állnak össze mondattá. Az eredményként kapott elemzési fa az elemzési címkékkel ellátott szavakat, a belőlük képzett csoportokat és a csoportok

hierarchiáját ábrázolja. Az elemzés minden tokenhez hozzárendeli a megfelelő elemzésifa-részlet zárójelekkel kódolt formáját.

Az emChunk eszköz azonosítja a szövegben a főnévi csoportokat (NP-ket), egész pontosan a maximális főnévi NP-ket, vagyis azokat, melyek nem részei magasabb szintű NP-nek. Itt az eddigiekkel ellentétben olyan annotációt adunk hozzá a szöveghez, mely több tokenre is kiterjedhet. Ezt a feladatot az eszköz, az alapját képező HunTag3 [8] szekvenciális tagger révén, kizárólag egyes tokenekre vonatkozó annotációk használatával oldja meg: minden tokenhez hozzárendel egy kódot, mely azt mondja meg, hogy az adott token az NP eleje (`B` kód), vége (`E` kód), közbülső eleme (`I` kód), vagy NP-n kívül esik (`O` kód), illetve külön jelet használ az egytokenes NP jelölésére (`1` kód). A többtokenes egységek ilyenfajta tokenenkénti annotációját nevezzük általánosságban IOB-típusú kódolásnak [9].

A szintén a HunTag3 rendszerre alapuló emNer tulajdonnév-felismerő eszköz a fentiekhez hasonló módon működik. Utolsó lépésként ez azonosítja és IOB kódolással megjelöli a szövegben található tulajdonneveket, ezenkívül besorolja őket az előre meghatározott névkategóriák valamelyikébe (személynév, intézménynév, földrajzi név, egyéb).

3. Kiegészítő eszközök

A fenti klasszikus szövegfeldolgozó eszközöket kiegészíti néhány olyan apróbb eszköz (ld. az 1. ábra szaggatott vonallal írt részét), ami az egész lánc hasznosságát, kényelmét növeli, könnyebben értelmezhetővé, felhasználhatóvá teszi az elemzések eredményét, az annotációkban lévő információt.

Az emMorph által szolgáltatott részletes morfológiai elemzés emberi fogyasztásra kevéssé alkalmas. A leírás olvashatóbbá tételét szolgálja a *ReadableMorphoAnalysis* nevű kiegészítő eszköz (2. ábra). Az `e-magyar.hu` honlap szövegelemző felületén is ezzel az eszközzel tesszük olvashatóbbá a részletes morfológiai elemzést.

```
amely[/N|Pro|Rel]=amely+ek[Pl]=ek+ről[Del]=ről
→
amely[/N|Pro|Rel] + ek[Pl] + ről[Del]
```

2. ábra. Az *amelyekről* szó emMorph szerinti morfológiai elemzése, és az elemzés könnyebben olvasható formája.

A magyarban az igekötő elválhat. Nyilván nem elvált (pl.: *elkészít*) és elvált (pl.: *készítette el*) esetben is ugyanarról az igekötős igéről van szó. Hasznos az igekötős igék összes alakját egyben látni, ezért jött létre a *PreverbIdentifier* kiegészítő eszköz, mely a függőségi elemzés alapján az igéhez kapcsolja a hozzá tartozó elváló igekötőt, és az igekötő és az igealak szótövének egybeírásaként megadja az igekötős szótövet – elváló esetben is.

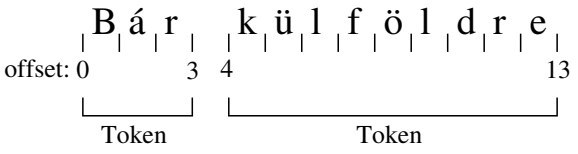
Ahogy említettük, az emChunk és az emNer IOB kódolással látja el a tokeneket. Ezt a kényelmesebb feldolgozhatóság érdekében az *Iob2Annot* kiegészítő eszköz önálló annotációvá: elkülönült egységeken lévő attribútumok sorozata helyett egy egységgé, a szöveg egy részletéhez rendelt 1 db címkévé alakítja.

A hasznos kiegészítő eszközök között említjük meg a GATE rendszerben eleve meglévő ún. *JAPE* transzdúcert is, mely az annotációk fölött megfogalmazott reguláris kifejezésekkel teszi lehetővé a szövegekben való szofisztikált keresést, vagy ha úgy tekintjük, új annotációk létrehozását.

4. A GATE annotációs modellje és a GATE-integráció

Az `e-magyar` rendszer szövegfeldolgozó eszközláncát alkotó, fentiekben áttekintett különféle modulok integrációját a GATE [10] (`https://gate.ac.uk`) nyelvfeldolgozó keretrendszerben valósítottuk meg. A Java nyelven implementált GATE előnye, hogy kényelmes módszert biztosít tetszőleges számú nyelvfeldolgozó eszköz (ún. *Processing Resource*) rendszerbe illesztésére. A másik fontos előnyös tulajdonsága az egyszerű, univerzális annotációs modell, melyre építve biztosítható a gördülékeny kommunikáció az egyes modulok között.

A feldolgozás legelején a szövegben a *karakterközök* kapnak egy sorszámot (ez az ún. *offset*), és onnantól kezdve *minden* annotáció kiterjedését egy offset-pár fejezi ki, mely az annotáció elejét és végét adja meg (3. ábra). Ez, a szöveget és az annotációt szétválasztó (standoff) annotációs modell lehetőséget biztosít arra, hogy az annotációk tetszőleges módon átfedjék egymást. Az annotációknak attribútumai is lehetnek. Az elemzés során hozzáadott információ közvetlenül konkrét annotáció (címke) formájában (pl.: az egyes tokenekhez rendelt *Token* annotáció) vagy az annotációk attribútumaiban (pl.: a *Token* annotáció *lemma* attribútuma, mely az egyértelműsített szótövet tartalmazza) kap helyet.



3. ábra. A GATE annotációs modellje. Minden karakterköz rendelkezik egy offset azonosítóval, az annotációk elejét és végét ezekkel az offsetekkel adjuk meg. A példában szereplő első *Token* annotáció 0-tól 3-ig tart, és a *Bár* szót ragadja meg.

A GATE az annotációkat annotációs halmazokban (*AnnotationSet*-ekben) tárolja. Ez azt jelenti, hogy az igényeknek megfelelően más-más halmazba téve bizonyos annotációkat teljesen elkülöníthetünk egymástól. Tipikus eset: ha HTML-fájl a kiinduló szöveganyagunk, akkor a GATE az eredeti HTML-annotációt

automatikusan feldolgozza, leválasztja, és eltárolja egy *Original markup* nevű annotációhalmazba, így lehetővé teszi, hogy a szöveganyagot az egyszerű szöveg inputtal megegyező módon dolgozzuk fel, a nyelvi elemzés során keletkező hozzáadott annotációkat pedig másik halmazba téve külön kezeljük.

A GATE azt is megengedi, hogy az attribútumok értéke nemcsak puszta szöveget, hanem tetszőleges Java objektumot (például stringek listáját) tartalmazzon. Látjuk, hogy az annotációs modell általánosabb, erősebb egy sima XML-szerű markupnál egyrészt az átfedés lehetősége, másrészt az annotációs halmazok, harmadrészt az attribútumokban megengedett adatszerkezetek miatt.

E modell használatával az annotációk függetlenek egymástól, nem zavarják egymást. Ez hasznos megoldás: így minden modul csak a számára releváns annotációt kell, hogy beolvassa, az eredményét pedig kiírhatja a megfelelő meglévő vagy újonnan létrehozott annotációba, attribútumba. Például: a tokenizáló *Token* és *SpaceToken* elemeket hoz létre a szavaknak és a szóközöknek megfelelően, a morfológiai elemző már csak a *Token*ek listáját fogja lekérni, ezen végzi el a morfológiai elemzést, a *SpaceToken*eket pedig érintetlenül hagyja. A tokenizálót követő elemző lépések tipikusan a tokenek bizonyos attribútumait felhasználva új token-attribútumokat hoznak létre. A modulok paraméterezhetők abban a tekintetben, hogy mely annotációkon dolgozzanak, ezzel a rendszer rugalmassága még tovább növelhető.

Az alapvető integrációs feladat tehát az volt, hogy minden modult alkalmassá tegyünk arra, hogy a bemenetét a GATE annotációs modelljének megfelelő formából tudja venni, és a kimenetét is ennek megfelelő formában prezentálja. Más szóval minden eszközhöz egy GATE-es wrappert kellett gyártani, ami a szükséges adatkonverziókat elvégzi a GATE-s formátum és a modul saját formátuma között. Kiegészítő feladat, hogy ha az egymástól független offset-alapú annotációk között kapcsolatot akarunk megadni, akkor azt explicit meg kell tenni. Jó példa erre az emNer által felismert tulajdonneveket és az őket alkotó tokeneket összekötő kapcsolat: itt az lett a megoldás – és ezt valósítja meg az *Iob2Annot* kiegészítő eszköz –, hogy a tulajdonnév annotáció egy attribútumában soroljuk fel listaként a tulajdonnevet alkotó tokenek azonosítóit.

Egy fontos, technikai kérdés volt, hogy milyen módon integráljuk a nem Java nyelven írt (emToken, emMorph, emChunk, emNer) eszközöket. Úgy döntöttünk, hogy az adott más nyelvű program binárisát vagy a más nyelvű szkriptet közvetlenül hívjuk meg. Ennek ellenére, megfelelő technikák alkalmazása révén a legtöbb esetben (ld. 7. rész) nincs szükség minden hívásnál az eszközök jelentős időigénnyel bíró újbóli inicializálására. Az elkészült elemzőlánc Linux és Windows operációs rendszeren futtatható.

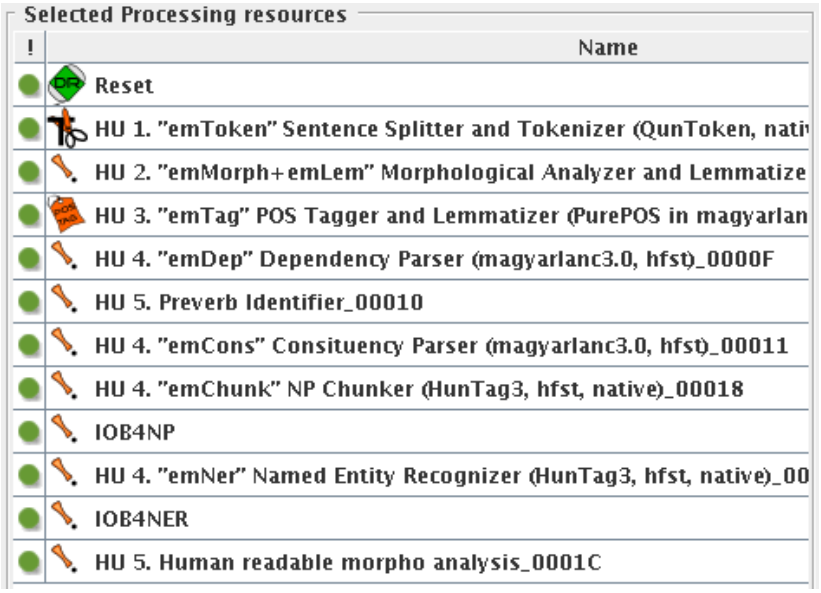
A működtetéshez szükség volt arra is, hogy az összes eszköz az új morfológiai elemző által adott kódkészlettel működjön. Ehhez elő kellett állítani a Szeged Treebank [11] új morfológiai kódokkal annotált változatát, majd ezen be kellett tanítani az emTag egyértelműsítőt, valamint a rá épülő eszközöket. A fenti feladatokat valósítottuk meg az integráció során.

A felhasználók számára előnyös, hogy a GATE annotációs modelljének és az integrációnak köszönhetően tehát nem kell törődni azzal, hogy: (1) hogyan

kell futtatni az egyes eszközöket, (2) milyen inputot várnak, és milyen outputot adnak az egyes eszközök, (3) egy-egy eszköz hogyan kezeli (használja fel, hagyja figyelmen kívül, őrzi meg) a meglévő annotációkat, milyen belső formátumot használ, és milyen annotációba helyezi el a saját elemzési eredményét. Az ilyen kérdéseket a GATE elrejti, elfedi, automatikusan megoldja. Az *Iob2Annot* pont egy egyszerű ilyenfajta GATE-es elrejtő megoldás: a HunTag3-alapú eszközök belső formátumának tekinthető IOB kódolást közvetlenül értelmezhető, szokásos önálló annotációvá alakítja.

5. Az elemzőlánc összeállítása a GATE Developerben

A GATE rendszer egyik fontos eleme a *GATE Developer* nevű grafikus felhasználói felület, ahol igényeink szerint állíthatjuk össze az elemzőláncokat az eszközökből (GATE Processing Resource-okból, PR), és lefuttathatjuk különféle szövegeken és a belőlük összeállított korpuszokon (GATE Language Resource-okon, LR). Ezek kívül számos kiegészítő funkció is rendelkezésre áll.



4. ábra. Az `e-magyar` szövegfeldolgozó eszközlánc a GATE Developer felületén.

A szövegfeldolgozó láncot tartalmazó *Lang_Hungarian* GATE plugin installálása után be kell tölteni az egyes eszközöket, és össze kell állítani belőlük a kívánt feldolgozó láncot. Először (1) jobb kattintás a bal panelen a *Processing Resources*-ra, és válasszuk ki a listából a kívánt eszközöket, majd (2) a bal panel

Applications részében hozzunk létre egy új `e-magyar` elnevezésű *Corpus Pipeline*-t, végül (3) a létrehozott *Corpus Pipeline*-ra kattintva állítsuk össze a fő panelen a láncot úgy, hogy a kívánt eszközöket a kívánt sorrendben a jobb oldali listába rendezzük. A teljes összeállított lánc a 4. ábrán látható.

Az ábrán a korábban leírt sorrendben szerepel az összes szövegfeldolgozó és kiegészítő eszköz. A számok arra utalnak, hogy hogyan épülnek egymásra az eszközök, hogy hányadik „rétegben" szerepel egy adott eszköz: 1. réteg az emToken, 2. réteg az egybeépített emMorph + emLem, 3. réteg az emTag, a 4. rétegben szintaktikai elemzők és a HunTag3-ra alapuló eszközök vannak, az 5. rétegben pedig a kiegészítő eszközök szerepelnek (vö: 1. ábra).

Az elemzés többszöri lefuttatása esetén hasznos, ha a lista elejére elhelyezünk egy *Document Reset PR*-t, ami minden futtatás előtt alaphelyzetbe állítja a dokumentumot, azaz törli az összes hozzáadott annotációt. Ezt a mindig rendelkezésre álló *ANNIE* pluginból tölthetjük be.

A legtöbb eszközt egyszerűen csak be kell töltenünk, és be kell tennünk az ábrán szereplő Corpus Pipeline-ba. Kivétel az *Iob2Annot*, melynek két példányára van szükségünk eltérő paraméterezéssel: egyszer a főnévi csoportok, másszor pedig a tulajdonnevek annotációjának átalakítására. Ahogy az ábrán látható, a két példányt értelemszerűen *IOB4NP* és *IOB4NER* elnevezéssel láttuk el. A paramétereket ezekre az eszközökre kattintva a képernyő alján állíthatjuk be az 1. táblázat szerint.

1. táblázat. Az *Iob2Annot* paraméterezése. A `inputIobAnnotAttrib` paraméter annak az attribútumnak a neve, amelyből a bemenő IOB annotációt veszi az eszköz, a `outputAnnotationName` pedig az új önálló annotáció neve. A megfelelő paraméterértékek a táblázatban láthatók.

	`inputIobAnnotAttrib`	`outputAnnotationName`
emChunk	`NP-BIO`	`NP`
emNer	`NER-BIO1`	`NE`

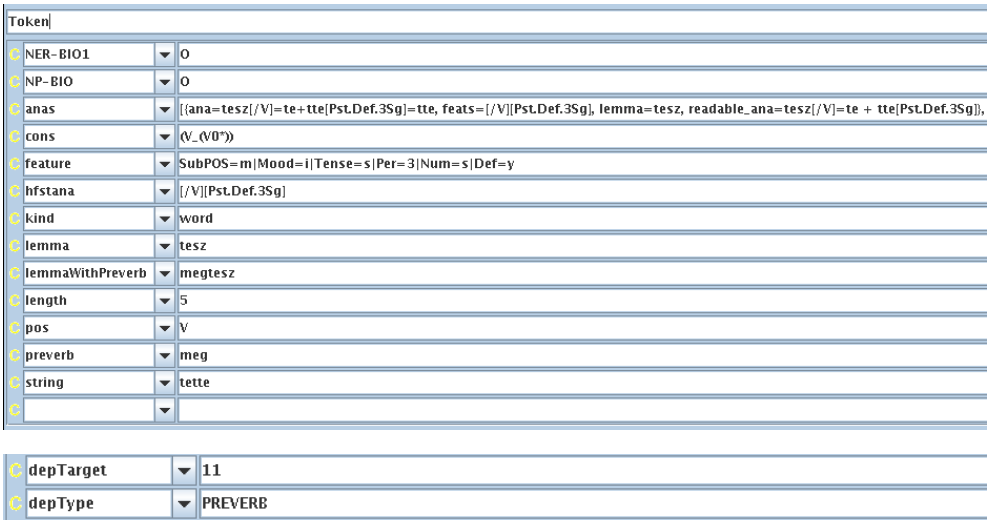
A paraméterező felületen igény szerint átállíthatjuk az eszközök alapbeállításait, ha például egy másik annotációs halmazba szeretnénk irányítani az elemzés eredményét.

6. Az elemzőlánc futtatása és az elemzés eredménye a GATE Developerben

Az összeállított elemzőlánc futtatásához (1) a bal panelen hozzunk létre egy *Language Resource*-ot: egy új *GATE Document*-et, ez fogja tartalmazni a feldolgozandó szöveget. A *GATE Developer* hatékonyan kezel számos formátumot (txt, HTML, XML, doc, stb.), belőlük automatikusan kinyeri a szöveges tartalmat. (2) Készítsünk a dokumentumból korpuszt: jobb kattintás a létrehozott

GATE Document-re, majd válasszuk a *New Corpus with this Document* lehetőséget. Végül (3) kattintsunk az `e-magyar` *Corpus Pipeline*-ra, a képernyő közepén, a *Corpus*-nál adjuk meg az imént létrehozott korpuszt, és kattintsunk lent a *Run this Application* gombra.

Az eredményeket a létrehozott *GATE Document*-re kattintva tekinthetjük meg az *Annotation Sets*, az *Annotation List*, és a kívánt annotált egység (*Token*, *NP*, *NE*) bekapcsolásával. A szövegben az egyes egységek fölé állítva az egeret megjelennek az adott egység attribútumainak értékei. Az elemzés eredményeként hozzáadott információ túlnyomó része a *Token* egységek attribútumaiként látható az 5. ábra és a 2. táblázat szerint.



5. ábra. Példamondatunk *tette* szavának attribútumai az `e-magyar` szövegfeldolgozó lánc lefuttatása után a GATE Developer felületén (fent). Részlet a példamondat *meg* szavának annotációjából a függőségi elemzés bemutatására (lent).

A *tette* szó természetesen nem része NP-nek, a *haladó* esetén a `NP-BIO` attribútumban `I-NP` szerepelne, ami azt jelenti, hogy a szó egy NP közbülső eleme.

A GATE Developerből az elemzett dokumentum GATE XML formátumban menthető el. Ez egy speciális XML formátum, amiben a GATE annotációs modellje ábrázolható. Tartalmazza a szöveget a szükséges offsetekkel együtt, és a szövegtől standoff módon elkülönítve az annotációkat attribútumaikkal. A kimentett XML-fájl pontosan a GATE Developer felületén is látható imént leírt annotációkat tartalmazza.

Itt a GATE Developernek csak a legalapvetőbb használatát mutattuk be, illetve a legszükségesebb információkat közöltük az `e-magyar` szövegfeldolgozó lánc

2. táblázat. A *Token* annotáció attribútumainak értelmezése. Az `anas` egy listában tartalmazza a részletes morfológiai elemzésekhez tartozó információkat, ezen belül: `ana` = részletes morfológiai elemzés, `feats` = emLem egyszerűsített elemzés, `lemma` = emLem szótő, `readable_ana` = `ana` olvashatóbb formája.

attribútum	értelmezés	a példában (5. ábra)
`NER-BIO1`	emNer IOB kód	a szó nem része tulajdonévnek
`NP-BIO`	emChunk IOB kód	a szó nem része NP-nek
`anas`	emMorph + emLem kimenet	
`cons`	emCons annotáció	egytagú igei csoport
`depTarget`	emDep szülő azonosítója	a *meg* szülője a *tette*
`depType`	emDep relációtípus	`PREVERB` (igekötő)
`feature`	az emTag kimenetéből a szintaktikai elemzők számára meghatározott jellemzők	a *tette* jegyei
`hfstana`	az emTag által választott elemzéshez tartozó egyszerűsített elemzés a HunTag3 eszközök számára	a *tette* szófaja és elemzése
`kind`	emToken szótípus	`word` (szó)
`lemma`	emTag szótő	*tesz*
`lemmaWithPreverb`	elváló igekötő esetén az igekötős szótő	*megtesz*
`length`	emToken szóhossz	5 karakter
`pos`	emTag szófaj	`V` (ige)
`preverb`	elváló igekötő esetén az igéhez tartozó igekötő	*meg*
`string`	emToken szóalak	*tette*

által szolgáltatott elemzés, annotáció értelmezéséhez. A GATE használatának további részletei és lehetőségei tekintetében a GATE rendszer dokumentációjára utalunk.

7. Négyféle hozzáférési, használati mód

A különböző felhasználói csoportok igényei szerint négy különböző módon lehet hozzáférni a rendszerhez, ezt tekintjük át az alábbiakban.

A legszélesebb érdeklődői kör számára lehetőség az `e-magyar.hu` honlap használata, mely a teljes elemzési láncot lefuttatja korlátozott szövegmennyiségen, az eredményt megjeleníti, letölthetővé teszi, a szintaktikai elemzések eredményét grafikusan is ábrázolja. Nem kell semmit installálni, az elemzés böngészőből futtatható, csupán annyi a teendő, hogy az elemzendő szöveget be kell másolni a honlap *Szövegelemző* felületére. A honlap a közoktatásban is használható demonstrációs eszköz, részletesebb leírás [1]-ben olvasható róla.

Komolyabb szövegelemzési feladathoz, digitális bölcsészeti kutatáshoz, ha az elemzőlánc bővítésére, új eszközök beépítésére van igény, illetve ha nagyobb mennyiségű az elemzendő szöveg, a GATE rendszer *GATE Developer* nevű grafikus felületének használata ajánlott, ahogy ezt a fentiekben bemutattuk. A GATE Developerben összeállítható a kívánt lánc, lefuttatható és az elemzések eredmény megjeleníthető. Az `e-magyar` elemzőláncon kívül megkapjuk a teljes GATE arzenált, a különféle létező nyelvfeldolgozó eszközeivel (az annotációtörlőtől a gépi tanulásig), és hasznos kiegészítő funkcióival, mint például a többféle bemeneti szövegformátum kezelése, az elemzőeszközök paraméterezhetősége, a kiértékelő modul, a kézi annotáló eszköz vagy a JAPE transzdúcer. Első futtatáskor lassabb működést tapasztalunk, mert ekkor töltődnek be a szükséges erőforrások, modellek, a statikus erőforrásoknak köszönhetően azonban a további futtatásoknál ez a plusz időigény nem jelentkezik. A GATE rendszer telepítése után csupán a GATE Developer saját egyszerű telepítési mechanizmusát kell használni, mely az általunk publikált GATE Plugin repozitóriumból letölti és beilleszti a rendszerbe a *Lang_Hungarian* plugint, mely a teljes láncot tartalmazza. Ennek leírása megtalálható az `e-magyar` szövegfeldolgozó lánc github repozitóriumában: `https://github.com/dlt-rilmta/hunlp-GATE`.

Jelentősebb méretű szöveganyag elemzéséhez a GATE parancssori hozzáférését az ún. *GATE Embedded* technológiát ajánljuk. Ennek révén beépíthetjük az eszközöket nagyobb szoftverrendszerekbe, vagy használhatjuk őket önállóan. A *pipeline*-ként aposztrofált megvalósításunk a GATE alapműködésének megfelelően – a használati módok közül egyetlenként – minden indításkor betölti az erőforrásokat, ezért használata csak nagyobb szövegmennyiség esetén célszerű.

A negyedik használati mód – az ún. *gate-server* – szintén a GATE Embedded technológiára épül, kliens-szerver architektúrában működik. A GATE-et egy HTTP szerverbe csomagoltuk, és kívülről, URL letöltésekkel használjuk. Így lehetővé válik az eszközök hatékony inicializálása: a GATE Developerhez hasonlóan ez a megoldás is csak egyszer tölti be az erőforrásokat, a szerver indításakor. Ez nagyon hasznos tulajdonság összevetve azzal az esettel, amikor az eszközöket a GATE-től függetlenül futtatnánk. A gate-server egyszerre kis méretű szövegdarabot dolgoz fel, nagy korpusz elemzése a korpusz feldarabolásával oldható meg. Egy gate-server üzemel az `e-magyar.hu` honlap mögött is. A harmadik és negyedik módszer leírása szintén az említett github repozitóriumban található, a használatba vételhez szükséges a github repozitórium klónozása.

8. Köszönetnyilvánítás

Az elemzőlánc integrációja az MTA 2015. évi Infrastruktúra-fejlesztési Pályázat 2. kategóriájában elnyert támogatás segítségével készült.

9. Konklúzió

Létrejött egy olyan magyar szövegfeldolgozó eszközlánc, mely egyesíti a legtöbb jelenleg elérhető nyílt forrású magyar szövegelemző eszközt. Tartalmaz egy új

Unicode-képes tokenizálót, valamint az eddigi magyar morfológiai elemzők jó tulajdonságait egyesítő jó minőségű új morfológiai elemzőt. Mindegyik eszköznek a legújabb verziója van beépítve, illetve mindegyik eszköz fel van készítve az új morfológiai kódkészlet használatára. Bízunk benne, hogy a különféle használati módoknak köszönhetően hasznos lesz nem csak a nyelvtechnológusok számára, nagy korpuszok elemzésére, hanem a kényelmes grafikus felhasználói felület révén a humán tudományok kutatói számára, illetve a honlap által a nagyközönség számára is. Reméljük, hogy a jövőben számos további újonnan létrehozott vagy akár korábban már meglévő magyar szövegfeldolgozó eszköz épül majd be ebbe a rugalmasan bővíthető keretrendszerbe, így egyre gazdagabb elemzési lehetőségek válnak elérhetővé. Ehhez lehetőségeinkhez mérten támogatást is nyújtunk.

Hivatkozások

1. Váradi, T., Simon, E., Sass, B., Gerőcs, M., Mittelholcz, I., Novák, A., Indig, B., Prószéky, G., Farkas, R., Vincze, V.: Az `e-magyar` digitális nyelvfeldolgozó rendszer. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017), Szeged: JATEPress (2017) (jelen kötetben)
2. Mittelholcz, I.: emToken: UTF-8 képes tokenizáló magyar nyelvre. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017), Szeged (2017) (jelen kötetben)
3. Novák, A., Siklósi, B., Oravecz, Cs.: A new integrated open-source morphological analyzer for Hungarian. In: Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC2016), Portorož (2016) 1315–1322
4. Lindén, K., Silfverberg, M., Pirinen, T.: HFST tools for morphology – an efficient open-source package for construction of morphological analyzers. In Mahlow, C., Piotrowski, M., eds.: State of the Art in Computational Morphology. Volume 41 of Communications in Computer and Information Science. Springer Berlin Heidelberg (2009) 28–47
5. Endrédy, I., Novák, A.: Szótövesítők összehasonlítása és alkalmazásaik. Alkalmazott Nyelvtudomány **XV**(1–2) (2015) 7–27
6. Orosz, Gy.: Hybrid algorithms for parsing less-resourced languages. PhD thesis, Roska Tamás Doctoral School of Sciences and Technology, Pázmány Péter Catholic University (2015)
7. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP. (2013) 763–771
8. Endrédy, I., Indig, B.: HunTag3: a general-purpose, modular sequential tagger – chunking phrases in English and maximal NPs and NER for Hungarian. In: 7th Language & Technology Conference (LTC '15), Poznań, Poland, Poznań: Uniwersytet im. Adama Mickiewicza w Poznaniu (2015) 213–218
9. Ramshaw, L.A., Marcus, M.P.: Text chunking using transformation-based learning. In: Proceedings of the 3rd Annual Workshop on Very Large Corpora. (1995) 82–94
10. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M.A., Saggion, H., Petrak, J., Li, Y., Peters, W.: Text Processing with GATE (Version 6). (2011)
11. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian Dependency Treebank. In: Proceedings of LREC 2010, Valletta, Malta, ELRA (2010)

emLam – a Hungarian Language Modeling baseline

Dávid Márk Nemeskey

Institute for Computer Science and Control
Hungarian Academy of Sciences
nemeskeyd@gmail.com

Abstract. This paper aims to make up for the lack of documented baselines for Hungarian language modeling. Various approaches are evaluated on three publicly available Hungarian corpora. Perplexity values comparable to models of similar-sized English corpora are reported. A new, freely downloadable Hungarian benchmark corpus is introduced.

1 Introduction

Language modeling (LM) is an integral part of several NLP applications, such as speech recognition, optical character recognition and machine translation. It has been shown that the quality of the LM has a significant effect on the performance of these systems [5, 7]. Accordingly, evaluating language modeling techniques is a crucial part of research. For English, a thorough benchmark of n-gram models was carried out by Goodman [10], while more recent papers report results for advanced models [20, 8]. Lately, the One Billion Word Benchmark corpus (1B) [8] was published for the sole reason of measuring progress in statistical language modeling.

The last decade saw dramatic advances in the field of language modeling. Training corpora grew from a few million words (e.g. the Brown corpus) to gigaword, such as 1B, while vocabulary size increased from a few 10k to several hundred thousands. Neural networks [3, 21, 19] overtook n-grams as the language model of choice. State-of-the-art LSTMp networks achieve up to 55% reductions in perplexity compared to 5-gram models [14].

Surprisingly, these developments left few traces in the Hungarian NLP literature. Aside from an interesting line of work on morphological modeling for speech recognition [23, 18], no study is known to the author that addresses issues of Hungarian language modeling. While quality works have been published in related fields, language model performance is often not reported, or is not competitive: e.g. in their otherwise state-of-the-art system, Tarján et al. [28] use a 3-gram model that achieves a perplexity of 400[1] on the test set — a far cry from the numbers reported in [8] and here.

In this paper, we mean to fill this gap in two ways. First, we report baselines for various language modeling methods on three publicly available Hungarian corpora. Hungarian poses a challenge to word-based LM because of its agglutinative nature. The proliferation of word forms inflates the vocabulary and decreases the number of contexts a word form is seen during training, making the data sparsity problem much more

[1] Personal communication with the author.

pronounced than it is for English. This makes it especially interesting to see how the performance of the tested methods translate to Hungarian.

Second, we present a version of the Hungarian Webcorpus [11] that can be used as a benchmark for LM models. Our motivation was to create the Hungarian equivalent of the One Billion Word Benchmark corpus for English: a freely available data set that is large enough to enable the building of high-quality LMs, yet small enough not to pose a barrier to entry to researchers. We hope that the availability of the corpus will facilitate research into newer and better LM techniques for Hungarian.

The software components required to reproduce this work, as well as the benchmark corpus, comprise the emLam module[2] of e-magyar.hu [30]. The scripts have been released as free software under the MIT license, and can be downloaded from the emLam repository[3].

The rest of the paper is organized as follows. The benchmark corpora, as well as our solution to the data sparsity problem is described in Section 2. In Section 3 we formally define the language modeling task and introduce the methods evaluated. Results are presented in Section 4. Finally, Section 5 contains our conclusions and ideas left for future work.

2 The Hungarian Datasets

We selected three publicly available Hungarian corpora for benchmarking. The corpora are of various sizes and domains, which enabled us to evaluate both small- and large-vocabulary LM configurations. The corpus sizes roughly correspond to those of the English corpora commonly used for LM benchmarks, making a comparison between the two languages easier.

The Szeged Treebank [31] is the largest manually annotated corpus of Hungarian. The treebank consists of CoNLL-style tsv files; we used a version in which the morphological features had been converted to KR codes to keep in line with the automatic toolchain described below. At around 1.5 million tokens, it is similar in size to the Penn Treebank [16], allowing us a direct comparison of small-vocabulary LM techniques.

The filtered version of the Hungarian Webcorpus [11] is a semi-gigaword corpus at 589m tokens. It consists of webpages downloaded from the `.hu` domain that contain an "acceptable number of spelling mistakes". The downloadable corpus is already tokenized; we further processed it by performing lemmatization, morphological analysis and disambiguation with Hunmorph [29]: `ocamorph` for the former two and `hunlex` for the latter.

The Hungarian Gigaword Corpus (MNSZ2) [25] is the largest public Hungarian corpus. At around 1G tokens, it is comparable in size to the English 1B corpus. We preprocessed the raw text with the same tools as above.

We decided to use the 'old' `hun*` tools because at the time of writing, the `e-magyar` toolchain was not yet production ready, and the version of the Szeged corpus that uses the new universal POS tags still contained conversion errors. Therefore, the results published here might be slightly different from what one can attain by running the scripts

[2] http://e-magyar.hu/hu/textmodules/emlam
[3] http://github.com/dlt-rilmta/emLam

in the `emLam` repository, should the issues above be addressed. However, any such differences will be, most likely, insignificant.

2.1 Preprocessing

As mentioned before, the main challenge of modeling an agglutinative language is the number of distinct word forms. The solution that works well for English — putting all word forms into the vocabulary — is not reasonable: on one hand, the vocabulary size would explode (see Table 1); on the other, there is a good chance the training set does not contain all possible word forms in the language.

The most common solution in the literature is to break up the words into smaller segments [12, 2, 4]. The two main directions are statistical and morphological word segmentation. While good results have been reported with the former, we opted for the latter: not only is it linguistically more motivated, it also ensures that the tokens we end up with are meaningful, making the LM easier to debug.

We ran the aforementioned pipeline on all words in the corpus, and split all inflectional prefixes (as well as some derivational ones, such as `<COMPAR>`, `<SUPERLAT>`) into separate tokens. Only inflections marked by the KR code are included; the default zero morphemes (the nominative case marker and the present-tense third person singular for verbs) are not. A few examples:

`jelmondatával` → `jelmondat <POSS> <CAS<INS>>`
`akartak` → `akar <PAST> <PLUR>`

One could say that by normalizing the text like this, we ”deglutenized” it; therefore, the resulting variants of the corpora shall be referred to as ”gluten-free” (GLF) from now on.

The full preprocessing pipeline is as follows:

1. Tokenization and normalization. The text was lowercased, converted to `utf-8` and and deglutenized
2. (Webcorpus only) Duplicates sentences were removed, resulting in a 32.5% reduction in corpus size.
3. Tokens below a certain frequency count were converted into `<unk>` tokens. The word distribution proved different from English: with the same threshold as in the 1B corpus (3), much more distinct tokens types remained. To be able to test LMs with a vocabulary size comparable to 1B, we worked with different thresholds for the two gigaword corpora: Webcorpus was cut at 5 words, MNSZ2 at 10. An additional thresholding level was introduced at 30 (50) tokens to make RNN training tractable.
4. Sentence order was randomized
5. The data was divided into train, development and test sets; 90%–5%–5% respectively.

Table 1 lists the main attributes of the datasets created from the three corpora. Where not explicitly marked, the default count threshold (3) is used. The corresponding English corpora are included for comparison. It is clear from comparing the raw and GLF

datasets that deglutenization indeed decreases the size of the vocabulary and the number of OOVs by about 50%. Although not shown in the table, this reduction ratio remains consistent among the various thresholding levels.

Also apparent is that, compared to the English corpora, the number of unique tokens is much bigger even in the default Hungarian GLF datasets. Preliminary inquiry into the data revealed that three phenomena account for the majority of the token types between the 3 and 30 (50) count marks: compound nouns, productive derivations and named entities (with mistyped words coming in at fourth place). Since neither the Szeged corpus, nor (consequently) the available morphological disambiguators take compounding and derivation into account, no immediate solution was available for tackling these issues. Therefore, we decided to circumvent the problem by introducing the higher frequency thresholds and concentrating on the problem of inflections in this study.

Dataset	Sentences	Tokens	Vocabulary	<unk>s	Analysis
Szeged	81 967	1 504 801	38 218	125 642	manual
Szeged GLF		2 016 972	23 776	55 067	
Webcorpus	26 235 007	481 392 824	1 971 322	5 750 742	automatic
Webcorpus GLF		683 643 265	960 588	3 519 326	
Webcorpus GLF-5		”	625 283	4 647 706	
Webcorpus GLF-30		”	185 338	9 393 015	
MNSZ2	44 329 309	624 830 138	2 988 629	11 614 583	automatic
MNSZ2 GLF		852 232 675	1 714 844	5 729 509	
MNSZ2 GLF-10		”	630 863	10 845 301	
MNSZ2 GLF-50		”	197 542	19 547 859	
PTB	49 199	1 134 978	10 000		manual
1B	30 607 716	829 250 940	793 471		automatic

Table 1. Comparison of the three Hungarian corpora

The preprocessing scripts are available in the emLam repository.

2.2 The Benchmark Corpus

Of the three corpora above, the Hungarian Webcorpus is the only one that is freely downloadable and available under a share-alike license (Open Content). Therefore, we decided to make not only the scripts, but the preprocessed corpus as well, similarly available for researchers.

The corpus can be downloaded as a list of tab-separated files. The three columns are the word, lemma and disambiguated morphological features. A unigram (word and lemma) frequency dictionary is also attached, to help create count-thresholded versions. The corpus is available under the Creative Commons Share-alike (CC SA) license.

Such a corpus could facilitate language modeling research in two ways. First, any result published using the corpus is easily reproducible. Second, the fact that it has been

preprocessed similarly to the English 1B corpus, makes comparisons such as those in this paper possible and meaningful.

3 Language Modeling

The task of (statistical) language modeling is to assign a probability to a word sequence $S = w_1, ..., w_N$. In this paper, we only consider sentences, but other choices (paragraphs, documents, etc.) are also common. Furthermore, we only concern ourselves with *generative* models, where the probability of a word does not depend on subsequent tokens. The probability of S can then be decomposed using the chain rule, as

$$P(S) = P(w_1, ..., w_N) = \sum_{i=1}^{N} P(w_i | w_1, ..., w_{i-1}). \quad (1)$$

The condition $(w_1, ..., w_{i-1})$ is called the *context* of w_i. One of the challenges of language modeling is that the number of possible contexts is infinite, while the training set is not. Because of this, the full context is rarely used; LMs approximate it and deal with the data sparsity problem in various ways.

In the following, we introduce some of the state-of-the-art methods in discrete and continuous language modeling.

3.1 N-grams

N-gram models work under the Markov assumption, i.e. the current word only depends on $n-1$ preceding words:

$$P(w_i | w_1, ..., w_{i-1}) \approx P(w_i | w_{i-n+1}, ..., w_{i-1}). \quad (2)$$

An n-gram model is a collection of such conditional probabilities.

The data sparsity problem is addressed by smoothing the probability estimation in two ways: *backoff* models recursively fall back to coarser ($n-1, n-2$, ...-gram) models when the context of a word was not seen during training, while *interpolated* models always incorporate the lower orders into the probability estimation.

A variety of smoothing models have been proposed over the years; we chose modified Kneser-Ney (KN) [15, 9] as our baseline, since it reportedly outperforms all other n-gram models [10]. We used the implementation in the SRILM [27] library, and tested two configurations: a pruned backoff (the default)[4] and, similar to [8], an unpruned interpolated model[5]. All datasets described in Table 1 were evaluated; in addition, we also tested a GLF POS model, where lemmas were replaced with their respective POS tags.

[4] `-kndiscount`

[5] `-kndiscount -gt1min 1 -gt2min 1 -gt3min 1 -gt4min 1 -gt5min 1 -interpolate1 -interpolate2 -interpolate3 -interpolate4 -interpolate5`

3.2 Class-based n-grams

Class-based models exploit the fact that certain words are similar to others w.r.t. meaning or syntactic function. By clustering words into classes C according to these features, a class-based n-gram model estimates the probability of the next word as

$$P(w_i|w_1, ..., w_{i-1}, c_1, ..., c_{i-1}) \approx P(w_i|c_i)P(c_{i-n+1}, ..., c_{i-1}). \quad (3)$$

This is a Hidden Markov Model (HMM), where the classes are the hidden states and the words are the observations. The techniques proposed for class assignment fall into two categories: statistical clustering [6, 17] and using pre-existing linguistic information such as POS tags [24]. In this paper, we chose the latter, as a full morphological analysis was already available as a by-product of deglutenization.

It is generally agreed that class-based models perform poorly by themselves, but improve word-based models when interpolated with them.

3.3 RNN

In the last few years, Recurrent Neural Networks (RNN) have become the mainstream in language modeling research [19, 20, 32, 14]. In particular, LSTM [13] models represent the state-of-the-art on the 1B dataset [14]. The power of RNNs come from two sources: first, words are projected into a continuous vector space, thereby alleviating the sparsity issue; and second, their ability to encode the whole context into their state, thereby ”remembering” much further back than n-grams. The downside is that it can take weeks to train an RNN, whereas an n-gram model can be computed in a few hours.

We ran two RNN baselines:

1. the Medium regularized LSTM setup in [32]. We used the implementation[6] in Tensorflow [1]
2. `LSTM-512-512`, the smallest configuration described in [14], which uses LSTMs with a projection layer [26]. The model was reimplemented in Tensorflow, and is available from the emLam repository.

Due to time and resource constraints, the first baseline was only run on the Szeged corpus, and the second only on the smallest, GLF-30 (50) datasets.

3.4 Language Model Evaluation

The standard metric of language model quality is *perplexity* (PPL), which measures how well the model predicts the text data. Intuitively, it shows how many options the LM considers for each word; the lower the better. The perplexity of the sequence $w_1, ..., w_N$ is computed as

$$PPL = 2^H = 2^{\sum_{i=1}^{N} -\frac{1}{N} \log_2 P(w_i|w_1,...,w_{i-1})}, \quad (4)$$

where H is the cross-entropy.

[6] https://github.com/tensorflow/tensorflow/tree/master/tensorflow/models/rnn/ptb

Language models typically perform worse when tested on a different corpus, due to the differences in vocabulary, word distribution, style, etc. To see how significant this effect is, the models were not only evaluated on the test split of their training corpus, but on the other two corpora as well.

4 Evaluation

The results achieved by the n-gram models are reported in Table 2–5. Table 2 lists the perplexities achieved by KN 5-grams of various kinds; the odd one out is POS GLF, where the limited vocabulary enabled us to create up to 9-gram models. For MNSZ2, the reported score is from the 7-gram model, which outperformed 8- and 9-grams.

Similar results reported by others on the PTB and 1B are included for comparison. A glance at the table shows that while word-based 5-grams performed much worse than their counterparts in English, the GLF-based models achieved similar scores.

While the perplexities of GLF models on Webcorpus and MNSZ2 are comparatively close, the perplexities of the word models are about 50% higher on Webcorpus. Finding the cause of this discrepancy requires further research. Two possible candidates are data sparsity (at the same vocabulary size, Webcorpus is 25% smaller) and a difference in the distribution of inflection configurations.

Corpus	Threshold	Word	GLF	Full POS	POS GLF
Szeged	3	262.77	123.66	35.20	22.90
Webcorpus	1	N/A	N/A	10.21	*6.05*
	5	328.22	67.90	N/A	N/A
	30	259.79	63.44	N/A	N/A
MNSZ2	1	N/A	N/A	11.88	*6.36*
	10	233.52	61.92	N/A	N/A
	50	174.65	55.53	N/A	N/A
PTB [22]	N/A	141.2			
1B [8]	3	90			

Table 2. 5-gram (*9 for POS GLF*) KN test results (PPL)

Table 3 shows the best n-gram perplexities achieved by GLF models. It can be seen that interpolated, unpruned models perform much better than backoff models.

Measuring class-based model performance led to surprising results. As mentioned earlier, the general consensus is that interpolating class- and word-based LMs benefit the performance of the latter; however, our findings (Table 4) did not confirm this. The class-based model could only improve on the unigram model, and failed to do so for the higher orders. The most likely explanation is that as the size of the vocabulary grows larger, the emission entropy increases, which is mirrored by the perplexity. This would explain why class-based n-grams seem to work on small corpora, such as the PTB, but not on MNSZ2.

Model	pruned backoff	unpruned interpolated
Szeged GLF	123.66	**116.32**
Webcorpus GLF-5	67.90	**58.62**
Webcorpus GLF-30	63.44	**54.42**
MNSZ2 GLF-10	61.92	**51.22**
MNSZ2 GLF-50	55.53	**46.24**
PTB [22]	141.2	N/A
1B [8]	90	**67.6**

Table 3. The best KN 5-gram results

Another point of interest is the diminishing returns of PPL reductions as the n-gram orders grow. While we have not experimented with 6-grams or higher orders, it seems probable that performance of GLF models would peak at 6- or 7-grams on MNSZ2 (and Webcorpus). For word-based models, this saturation point arrives much earlier: while not reported in the table, the perplexity difference between 4- and 5-gram models is only 1-2 point. This implies that GLF models are less affected by data sparsity.

Model	GLF-10	POS → GLF-10	GLF-50	POS → GLF-50
1-gram	2110	653.67	2175	568.53
2-gram	127.17	327.74	115.38	285.72
3-gram	84.81	294.20	73.31	256.60
4-gram	66.06	274.70	59.41	239.64
5-gram	61.92	261.79	55.53	228.40

Table 4. Class (POS)-based model performance on the MNSZ2

It is a well-known fact that the performance of LMs degrade substantially when they are not evaluated on the corpus they were trained on. This effect is clearly visible in Table 5. It is also evident, however, that GLF datasets suffer from this problem to a much lesser extent: while the perplexity more than doubled for the word-based MNSZ2 LMs, it only increased by 50–60% for GLF models. A similar effect can be observed between the full and GLF POS models.

Interestingly, the Webcorpus word models exhibit the smallest perplexity increase of 10-15%. Contrasting this result with Table 2 seems to suggest that there exists a trade-off between predictive power and universality. However, it is worth noting that the performance of these word models still lags well behind that of GLF models.

Finally, Table 6 reports the perplexities achieved by the RNN models. Two conclusions can be drawn from the numbers. First, in line with what has been reported for English by many authors, RNNs clearly outperform even the best n-gram models. Second, the numbers are similar to those reported in the original papers for English. This,

Model	Evaluated on	1 tokens	5 (10) tokens	30 (50) tokens
Webcorpus word	MNSZ2		377.88	291.98
MNSZ2 word	Webcorpus		566.60	397.13
Webcorpus GLF	MNSZ2		109.71	94.59
MNSZ2 GLF	Webcorpus		92.51	84.91
Webcorpus Full POS	MNSZ2	16.14		
MNSZ2 Full POS	Webcorpus	16.49		
Webcorpus POS GLF	MNSZ2	8.35		
MNSZ2 POS GLF	Webcorpus	7.73		

Table 5. Cross-validation results between Webcorpus and MNSZ2 with various thresholds.

together with similar observations above for n-grams, proves that once the ”curse of agglutination” is dealt with, a GLF Hungarian is no more difficult to model than English.

Model	Dataset	Perplexity
Medium regularized	Szeged GLF	35.20
LSTM-512-512	Webcorpus GLF-30	40.46
LSTM-512-512	MNSZ2 GLF-50	38.78
Medium regularized [32]	PTB	82.07
LSTM-512-512 [14]	1B	54.1

Table 6. LSTM model performance

5 Conclusion

This work contributes to Hungarian language modeling in two ways. First, we reported state-of-the-art LM baselines for three Hungarian corpora, from million to gigaword size. We found that raw, word-level LMs performed worse than they do for English, but when the text was split into lemmas and inflectional affixes (the ”gluten-free” format), results were comparable to those reported on similar-sized English corpora.

Second, we introduced a benchmark corpus for language modeling. To our knowledge, this is the first such dataset for Hungarian. This specially prepared version of the Hungarian Webcorpus is freely available, allowing researchers to easily and reproducibly experiment with new language modeling techniques. It is comparable in size to the One Billion Word Benchmark corpus of English, making comparisons between the two languages easier.

5.1 Future Work

While the methods reported here can be called state-of-the-art, many similarly effective modeling approaches are missing. Evaluating them could provide additional insight into how Hungarian "works" or how Hungarian and English should be modeled differently. Understanding the unusual behaviour of word models on Webcorpus also calls for further inquiry into language and corpus structure.

The performance of the models here was measured in isolation. Putting them into use (maybe with some adaptation) in NLP applications such as ASR or ML could answer the question of whether the reduction in perplexity translates to similar reductions in WER or BLEU.

The most glaring problem touched upon, but not addressed, in this paper, is the effect of compounding and derivation on vocabulary size. A way to reduce the number of words could be a more thorough deglutenization algorithm, which would split compound words into their parts and strip productive derivational suffixes, while leaving frozen ones such as `ház·as·ság` untouched. This could indeed be a case when a gluten free diet does make one slimmer.

Acknowledgements

This work is part of the `e-magyar` framework and was supported by the Research Infrastructure Development Grant, Category 2, 2015 of the Hungarian Academy of Sciences.

References

[1] Martın Abadi et al. "TensorFlow: Large-scale machine learning on heterogeneous systems, 2015". In: *Software available from tensorflow. org* 1 (2015).

[2] Mohamed Afify, Ruhi Sarikaya, Hong-Kwang Jeff Kuo, Laurent Besacier, and Yuqing Gao. "On the use of morphological analysis for dialectal Arabic speech recognition." In: *INTERSPEECH*. 2006, pp. 277–280.

[3] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. "A Neural Probabilistic Language Model". In: *Journal of Machine Learning Research* 3 (2003), pp. 1137–1155. URL: `http://www.jmlr.org/papers/v3/bengio03a.html`.

[4] Jan A Botha and Phil Blunsom. "Compositional Morphology for Word Representations and Language Modelling". In: *ICML*. 2014, pp. 1899–1907.

[5] Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. "Large Language Models in Machine Translation". In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 858–867. URL: `http://www.aclweb.org/anthology/D/D07/D07-1090`.

[6] P.F. Brown, V.J. Della Pietra, P.V. de Souza, J.C. Lai, and R.L. Mercer. "Class–based n–gram models of natural language". In: *Computational Linguistics* 18.4 (1992), pp. 467–480.

[7] Ciprian Chelba, Dan Bikel, Maria Shugrina, Patrick Nguyen, and Shankar Kumar. *Large Scale Language Modeling in Automatic Speech Recognition*. Tech. rep. Google, 2012. URL: `https://research.google.com/pubs/pub40491.html`.

[8] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. "One billion word benchmark for measuring progress in statistical language modeling". In: *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*. 2014, pp. 2635–2639.

[9] Stanley F Chen and Joshua Goodman. *An empirical study of smoothing techniques for language modeling*. Tech. rep. TR-10-98. Cambridge, MA: Computer Science Group, Harvard University, Aug. 1998, p. 63.

[10] Joshua T. Goodman. "A bit of progress in language modeling". In: *Computer Speech & Language* 15.4 (2001), pp. 403–434.

[11] Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. "Creating open language resources for Hungarian". In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*. ELRA, 2004, pp. 203–210.

[12] Teemu Hirsimäki, Mathias Creutz, Vesa Siivola, and Mikko Kurimo. "Morphologically Motivated Language Models in Speech Recognition". In: *Proceedings of AKRR'05, International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*. Espoo, Finland: Helsinki University of Technology, Laboratory of Computer and Information Science, June 2005, pp. 121–126. URL: `http://www.cis.hut.fi/AKRR05/papers/akrr05tuulos.pdf`.

[13] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (Nov. 1997), pp. 1735–1780.

[14] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. "Exploring the limits of language modeling". In: *arXiv preprint arXiv:1602.02410* (2016).

[15] Reinhard Kneser and Hermann Ney. "Improved backing-off for m-gram language modeling". In: *International Conference on Acoustics, Speech, and Signal Processing, 1995. ICASSP-95.* Vol. 1. IEEE. 1995, pp. 181–184.

[16] Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. "Building a Large Annotated Corpus of English: The Penn Treebank". In: *Computational Linguistics* 19 (1993), pp. 313–330.

[17] Sven Martin, Jörg Liermann, and Hermann Ney. "Algorithms for bigram and trigram word clustering". In: *Speech communication* 24.1 (1998), pp. 19–37.

[18] Péter Mihajlik, Zoltán Tuske, Balázs Tarján, Bottyán Németh, and Tibor Fegyó. "Improved recognition of spontaneous Hungarian speech — Morphological and acoustic modeling techniques for a less resourced task". In: *IEEE Transactions on Audio, Speech, and Language Processing* 18.6 (2010), pp. 1588–1600.

[19] Tomas Mikolov. "Statistical Language Models Based On Neural Networks". PhD thesis. Faculty of Information Technology, Brno University of Technology, 2012.

[20] Tomas Mikolov, Anoop Deoras, Stefan Kombrink, Lukas Burget, and Jan Cernockỳ. “Empirical Evaluation and Combination of Advanced Language Modeling Techniques.” In: *INTERSPEECH*. 2011, pp. 605–608.

[21] Tomáš Mikolov, Anoop Deoras, Daniel Povey, Lukáš Burget, and Jan Černockỳ. “Strategies for training large scale neural network language models”. In: *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE. 2011, pp. 196–201.

[22] Tomas Mikolov and Geoffrey Zweig. “Context dependent recurrent neural network language model”. In: *SLT*. 2012, pp. 234–239.

[23] Bottyán Németh, Péter Mihajlik, Domonkos Tikk, and Viktor Trón. “Statisztikai és szabály alapú morfológiai elemzők kombinációja beszédfelismerő alkalmazáshoz”. In: *Proceedings of MSZNY 2007*. Szegedi Tudományegyetem, Nov. 2007, pp. 95–105.

[24] Thomas R Niesler, Edward WD Whittaker, and Philip C Woodland. “Comparison of part-of-speech and automatically derived category-based language models for speech recognition”. In: *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*. Vol. 1. IEEE. 1998, pp. 177–180.

[25] Csaba Oravecz, Tamás Váradi, and Bálint Sass. “The Hungarian Gigaword Corpus”. In: *Proceedings of LREC 2014*. 2014.

[26] Hasim Sak, Andrew W Senior, and Françoise Beaufays. “Long short-term memory recurrent neural network architectures for large scale acoustic modeling.” In: *INTERSPEECH*. 2014, pp. 338–342.

[27] Andreas Stolcke, Jing Zheng, Wen Wang, and Victor Abrash. “SRILM at sixteen: Update and outlook”. In: *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*. Vol. 5. 2011.

[28] Balázs Tarján, Ádám Varga, Zoltán Tobler, György Szaszák, Tibor Fegyó, Csaba Bordás, and Péter Mihajlik. “Magyar nyelvű, élő közéleti- és hírműsorok gépi feliratozása”. In: *Proc. MSZNY 2016*. Szegedi Tudományegyetem, 2016, pp. 89–99.

[29] Viktor Trón, Gyögy Gyepesi, Péter Halácsky, András Kornai, László Németh, and Dániel Varga. “Hunmorph: Open Source Word Analysis”. In: *Proceedings of the ACL Workshop on Software*. Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 77–85.

[30] Tamás Váradi et al. “`e-magyar`: digitális nyelvfeldolgozó rendszer”. In: *XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017)*. Szeged, 2017, (this volume).

[31] Veronika Vincze, Viktor Varga, Katalin Ilona Simkó, János Zsibrita, Ágoston Nagy, Richárd Farkas, and János Csirik. “Szeged Corpus 2.5: Morphological Modifications in a Manually POS-tagged Hungarian Corpus”. In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), May 2014. ISBN: 978-2-9517408-8-4.

[32] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. “Recurrent neural network regularization”. In: (2014). arXiv: `1409.2329 [cs.NE]`.

e-Magyar beszédarchívum

Kornai András[1], Szekrényes István[2]

[1] MTA Nyelvtudományi Intézet,
1068 Budapest, Benczur u. 33., e-mail: andras@kornai.com
[2] Debreceni Egyetem, Általános és Alkalmazott Nyelvészeti Tanszék
4032 Debrecen, Egyetem tér 1, e-mail: szerkenyes.istvan@arts.unideb.hu

Kivonat Cikkünkben az e-magyar digitális nyelvfeldolgozó rendszer részeként létrejött nyílt forráskódú és szabad felhasználású beszédarchívum jelenlegi állapotáról és további terveiről számolunk be.

Kulcsszavak: beszédtechnológia, beszédarchívum, e-magyar

1. Céljaink

A beszédarchívum[3] létrehozásával három fő célunk volt. Az első és legfontosabb a magyar beszédtechnológiára annak kezdetei óta jellemző zárt kutatási és publikációs modell felváltása egy szabad, nyílt forrású (Free and Open Source Software, FOSS) modellel. Második célunk a hagyományos, gondosan felcímkézett és mind artikulációsan mind akusztikailag tiszta adatokon alapuló felügyelt tanulási módszerek felváltása gyengén felügyelt illetve felügyeletlen (weakly supervised, unsupervised) módszerekkel. Harmadik, az első kettőtől nem mindig könnyen elválasztható célunk pedig egy a digitális bölcsészeti munkát, elsősorban a szociológiát, történelemtudományt, folklorisztikát, és néprajzot beszédtechnológiai oldalról támogató platform alapjainak megteremtése.

2. Kiinduló állapot

Az `e-magyar` pályázat a nyelvtechnológiában, különösen a szószintű eszközök (morfológiai elemzés és generálás), de kisebb részben már a frázis- és mondatszintű eszközök területén teljessé tette a nyílt forrású adatok és eszközök bevezetését (Várad et al, ugyane kötetben), ennek minden, a fejlődést katalizáló előnyével együtt. Ez csak úgy volt lehetséges, hogy az évtizedek során komoly FOSS eszközök halmozódtak fel, melyek közül a teljesség igénye nélkül kiemeljük a `Hun*` és a `Magyarlánc` eszközláncokat, a monolinguális `Webkorpuszt` és a `Hunglish` párhuzamus korpuszt. Mostani projektünk elkezdése előtt a magyar beszédtechnológia szabadon letölthető adatokat nem tett közzé (az egyedi mérlegelésen alapuló hozzáférés-engedélyezést nem sorolhatjuk a FOSS paradigmába) sem a világszerte közismert beszédtechnológiai eszközök magyar honosításai nem voltak elérhetőek, annak ellenére, hogy a létező szoftverek, különösen a beszédfelismerés terén, elsősorban ilyeneken alapultak (ennek pontos mértéke természetesen csak a szoftverek nyilvánosságra kerülésével lesz megállapítható).

[3] `http://e-magyar.hu/hu`

3. A projekt eredményei

Elmondhatjuk, hogy a FOSS beszédarchívum megjelenésével a helyzet gyökeresen megváltozott. Az adatok szintjén elérhetővé vált sok ezer órányi jogtiszta adásminőségű (broadcast quality) és sokszáz órányi ennél rosszabb (communication quality) anyag. Ezeknél sokkal jobb minőséget képvisel a `BEA` spontánbeszéd-adatbázis [2], de kisebb, és nem teljesen FOSS. Hangsúlyoznánk, hogy a korszerű beszédtfelismerésben a jobb akusztikai minőség nem követelmény, sőt, immár több évtizedes tapasztalat, hogy a legjobban azok a beszédfelismerő rendszerek teljesítenek, melyeket reális, az alkalmazásban valóban fellépő akusztikai körülményeket tükröző adatokon tanítottak be.

Ugyanilyen változást hozott a projekt a követő szoftverek terén is. Több tucatnyi alternatíva telepítésével és összemérésével választottuk ki a legjobbakat. Számos okból utasítottunk el szoftvereket:

- Egzotikus nyelvet igényel (pl. Luá-t mint a corona[4])
- Elöregedett modulokat használ (pl. tcl/tk-t mint a snack[5])
- Zárt modulokat használ (pl. a pysonic[6])
- Rendszerspecifikus (leggyakrabban Windows)
- Dokumentálatlan (pl. a RawAudioSocket)
- Csak kutatásra használható (pl. az OpenSmile[7])
- Elhagyott (pl. a LiUM[8])
- Fontos formátumokat nem támogat (pl. az AudioLazy)

Tucatjával találtunk olyan szoftvereket, melyek egyszerre több szempontból is problematikusak, és van még egy pár olyan, amivel változatlanul próbálkozunk, ilyen pl. a `bob.bio.spear`[9] és a `Brno phoneme recognizer`[10].

A hangformátumok konverziójára végül a `SoX` és `ffmpeg` eszközöket, a beszédaktivitás detektálására és naplózás (diarization) céljára a `shout` programot (ld. 4.1), végül statisztikai nyelvmodellezésre az `srilm` eszközt (Nemeskey, ugyane kötetben) használtuk fel. Ez utóbbihoz olyan modelleket tettünk elérhetővé, melyek perplexitása 56, tudtunkkal az összes publikált (de le azért nem tölthető) modell perplexitását lényegesen megjavítva. Eredeti vállalásunkkal ellentétben *nem készült el*, de terveink között változatlanul szerepel az automatikus nyelvazonosítást lehetővé tevő szoftver.

[4] `https://docs.coronalabs.com/api/library/audio/play.html`
[5] `http://www.speech.kth.se/snack`
[6] `http://pysonic.sourceforge.net`
[7] `http://audeering.com/research/opensmile`
[8] `http://www-lium.univ-lemans.fr/diarization/doku.php/download`
[9] `https://pypi.python.org/pypi/bob.bio.spear/2.0.4`
[10] `http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context`

4. Új modulok integrációja

4.1. emDia, emSad

Az `emDia` beszélő diarizáló modul a 'ki, mikor beszélt' kérdésre ad választ (tehát a beszélőváltásokat állapítja meg), ez a nyílt forráskódú (GPL), C++-ban írt `SHOUT` Speech recognition toolkit [4] 'shout_segment' és 'shout_cluster' programjainak a használatával történik. A modul a bemeneti audio fájlt a `SoX` (Sound Exchange, GPL) eszközt használva konvertálja, így minden olyan formátumot elfogad, amit ez kezel (pl. mp3, wav). A diarizáló modul kimenete két RTTM (Rich Transcription Time Marked) kompatibilis fájl, amelyek a megtalált beszéd-zaj-csend, illetve a különböző beszélőkhöz tartalmazó audio szegmenseket írják le.

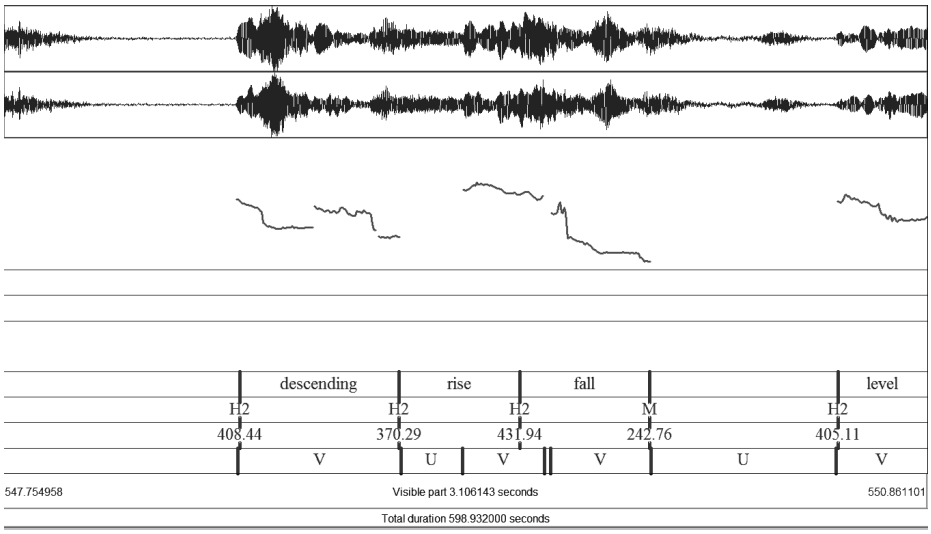
Az `emSad` modul a diarizáló modul első lépésének, a beszédtevékenység detekciónak az önálló futtatását teszi lehetővé. Szintén a `SoX` eszköz felhasználásával többféle bemeneti formátumot támogat. A modul funkciói közé tartozik még az azonos típusú szegmensek egyetlen hangfájllá konvertálása, ami pl. alkalmas egy beszédet, zajt és csendet vegyesen tartalmazó fájlból a beszéd kinyerésére.

4.2. emPros

Az `emPros` (eredeti nevén: `ProsoTool`) egy a `Praat` beszédfeldolgozó program [1] szkript nyelvén implementált, az élőnyelvi kommunikációban előforduló verbális megnyilatkozások prozódiájának elemzésére és lejegyzésére szolgáló algoritmus, amely a `HuComTech` projekt alapkutási céljai [5] érdekében került (gépi annotálást végző, offline eszközként) kifejlesztésre. A fejlesztés kezdeti szakaszának – még csak a terveket és a lehetőségeket feltáró – részeredményei a VIII. Magyar Számítógépes Nyelvészeti Konferencián kaptak először nyilvánosságot [10]. A későbbi publikációk elsősorban a beszéddallam automatikus lejegyzésére szolgáló, az `e-magyar`[11] projekt weboldalán is elérhető modul hátterét [9] és működését [8] tárgyalják. A további tervek között szereplő, a beszéd hangerőváltozásait és tempóját elemző modulok jelenleg is fejlesztés alatt állnak. Az algoritmus tesztelése a `Langua Archive`[12] és a `Meta-Share`[13] projekteken keresztül kutatási célokra közzétett `HuComTech korpusz`[14] magyar nyelvű, formális és informális dialógusokat rögzítő hangfelvételeinek és szöveges átiratainak felhasználásával, a korpusz széleskörű elemzési szempontokat átfogó annotációnak további bővítése céljából történt. A legfrissebb (eddig nem publikált) javítások és átdolgozások, melyeknek a program jelenlegi flexibilitását köszönheti, a `SegCor` projekt[15] közreműködésével, a `FOLK` korpusz [7] német nyelvű, változatos kondíciók között (2–14 adatközlővel) készített hangfelvételeinek elemzése során valósultak meg.

[11] `http://e-magyar.hu/hu/speechmodules/emPros`
[12] `https://tla.mpi.nl/`
[13] `http://metashare.nytud.hu/`
[14] `https://hdl.handle.net/1839/00-0000-0000-001A-E17C-1@view`
[15] `http://www1.ids-mannheim.de/prag/muendlichekorpora/segcor.html`

Az alkalmazás fejlesztését leginkább Piet Mertens szintén a `Praat` program szkript nyelvén, `Prosogram`[16] néven implementált, a tonális kontúrok pszichoakusztikai alapokon [3] történő stilizálását végző eljárása inspirálta [6], de az alaphang modulációinak kategorizálására használt módszereket tekintve a `Tilt`[17] intonációs modell paramétereiből is merített. Fontos különbség, hogy az intonáció elemzését az `emPros` a beszéd szegmentális szerkezetétől függetlenül, nem a szótagok szintjén végzi, így nem is igényli a szótaghatárok előzetes detektációját. A szegmentáció alapját az alapfrekvencia kontúr (a `Praat` program beépített funkcióival történő) simítása és stilizálása eredményeként kapott, a percepció számára nem releváns mikro-intonációs mozgásokat a beszéd hosszabb egységein átívelő intonációs trendekben integráló dallammenetek képezik. A dallammenetek kategorizálása és címkézése azok időtartama és Hertzben mérhető „amplitúdója" alapján történik, amely a vizsgált beszélő öt részre felosztott hangterjedelmével és átlagos hangmagasság ingadozásával kerül összevetésre.



1. ábra. A ProsoTool kimenete a Praat program szerkesztő felületén

Mivel a szkript beszélőnként végzi az intonáció elemzését, a hangfelvétel mellett egy olyan (Praat TextGrid formátumú) annotáció is bemeneti követelmény, amely a megnyilatkozások időbeli pozícióját beszélőnként külön tengelyen (annotációs szinten) tartalmazva reprezentálja a fordulóváltások akusztikai szerkezetét. Az `e-magyar` beszédfeldolgozó moduljai között helyett kapó `emDia` pontosan a föntebbi információkat szolgáltatja kimenetként, így az `emPros` a beszélő diarizáló kimenetén alkalmazott eljárásként integrálható, amelyben a beszélők hangjának izolált akusztikai elemzését egy a prozódiai moduloktól különválasz-

[16] `http://bach.arts.kuleuven.be/pmertens/prosogram/`
[17] `http://www.cstr.ed.ac.uk/projects/speech_tools/manual-1.2.0/c16909.htm`

tott előfeldolgozó algoritmus készíti elő. A kimenet a bemenetben jelölt beszélők szerint elkülönítve, `Praat TextGrid` formátumban kódolja a hanglejtés dallammenetekre szegmentált elemzését. A lejegyzés négy, a(z) 1. ábrán is látható, időben párhuzamos szintből áll. Az első szint a stilizálás eredményeként kapott dallammeneteket a „rise" (szökő), „fall" (lebegő), „ascending" (emelkedő), „descending" (ereszkedő), „level" (szinttartó) kategóriák valamelyikébe sorolja. A második szint a dallammenetek mozgását a beszélő 5 szintre ($L_2 < L_1 < M < H_1 < H_2$) felosztott hangterjedelmében pozícionálja. A harmadik szint az előző szint relatív értékeihez az eredeti, Hertzben mért értékeket társítja hozzá. A negyedik szint pedig a beszéd zöngés („V") és zöngétlen („U") szakaszait különíti el.

5. Együttműködésben várható eredmények

Az archívum bővülése több irányból is várható anélkül, hogy ez újabb anyagi vagy emberi erőforrásokat igényelne. Támogatásukról biztosítottak a NAVA, az OGyK, az OSzK, az MTA TK, Kisebbségkutató, és Szociológiai intézetek és más intézmények is, sőt az intézmények egy részétől már kaptunk is anyagokat.

Különösen fontos a hazai és környező országokbeli társadalomtudósok támogatása. A teljesség igénye nélkül: Havas Gábor, Lengyel Gabriella, Németh Szilvia, Zolnay János, Virág Tünde; a kolozsvári kisebbségkutató (Fosztó László, Kiss Tamás, Vitos Katalin, Lőrincz József), a marosvásárhelyi Sapientia (Gagyi József), a kolozsvári Kriza Társaság (Szabó Töhötöm), a Babes-Bolyai Egyetem (Tánczos Vilmos, Pozsony Ferenc), a kolozsvári, marosvásárhelyi rádiók anyagai (Maksay Ágnes, Tibád Zoltán).

Külön említést igényel Molnár Gusztáv hatalmas interjúanyaga (mintegy (70 óra, nagyrészt magyarul, de több mint 20 óra románul) a XX század olyan jelentős személyeivel mint Balogh Edgár vagy Szabó T. Attila. Sajnos ezen anyagok nagy része ma még kazettán van, de ezek átjátszását folyamatosan végezzük.

Különösebb plusz befektetés nélkül, csupán a meglevő folyamatok folytatásával az archívum még éveken át bővülni fog.

6. A továbblépés főbb irányai

Számítunk a közösség támogatására abban, hogy a beszédarchívum még jobban használható legyen. Az első és legfontosabb lépés ebben egy *adatkezelési* modell (data curation model) kialakítása kell legyen.

Kik adják az adatokat? A google kérdőív[18] kitöltésével bárki, aki szeretné adatait nyilvánosan hozzáférhetővé tenni.

[18] `https://docs.google.com/forms/d/e/1FAIpQLSdwBoeLh_g2A6FO5VbKONGIBYJ-CfWb83KXFClVodr68Bhm5w/viewform?c=0&w=1`

Kik őrzik az adatokat? Ennek infrastrukturális hátterét legalább 10 évre megadta az `e-magyar` finanszírozású hardver-fejlesztés, a szervezeti hátteret biztosítja az MTA Nyelvtudományi Intézet és az MTA SZTAKI közti megállapodás. Természetesen teljes idejű, vagy akár részidejű digitális könyvtáros felvétele a folyamatot nagyban gyorsítaná, erre azonban a pályázat egyszeri jellege nem adott módot.

Milyen metaadatokat tároljunk, és milyen sémában? A rendszer rugalmas, itt elsősorban az érdekelt felhasználók véleményét várjuk ahhoz, hogy igényeiknek a leginkább megfelelő adatbázis-sémát és keresési eszközöket illesszünk az adatokhoz. Terveink szerint ez nem kézi címkézéssel nyert „gold", hanem az `emSad`, az `emDia`[19], és a `emPros` az egész adaton való átfuttatásával keletkező „silver" adatokon fog alapulni.

A második kérdés a további szoftverek fejlesztése. Mint az `emPros` (ProsoTool)[20] példája mutatja, független `github` szoftver-repozitórium minden nehézség nélkül kapcsolható az e-magyar-hoz, és nagy örömmel várjuk a többi FOSS szoftver megjelenését.

7. Köszönetnyilvánítás

Köszönettel tartozunk Uwe Reichelnek és Mády Katalinnak (NYTI), továbbá a `speech@lists.mokk.bme.hu` levelezőlista minden tagjának számos hasznos ötletért és tanácsért, Pajkossy Katalinnak és Ács Juditnak (BME) az `emDia` és az `emSad` beüzemeléséért, Takács Dávidnak (Meltwater) és Gerőcs Mátyásnak (NYTI) a webes arculatért. Külön köszönet Schreiner Józsefnek (interNetWire Communications) a határon túli kutatások anyagának áttekintéséért és a digitalizáció beindításáért, és Both Zsoltnak (SZTAKI) a hardver beüzemeléséért.

Az `e-magyar` eszközlánc az MTA 2015. évi Infrastruktúra-fejlesztési Pályázat 2. kategóriájában elnyert támogatás segítségével valósult meg.

Hivatkozások

1. Boersma, Paul & Weenink, D.: Praat: doing phonetics by computer [computer program]. version 6.0.22. `http://www.praat.org/` (2016), retrieved 15 November 2016
2. Gósy, M. (ed.): Beszéd, adatbázis, kutatások. Akadémia (2012)
3. Hart, J.t.: Psychoacoustic backgrounds of pitch contour stylisation. IPO-APR 11, 11–19 (1976)
4. Huijbregts, M.: Segmentation, diarization and speech transcription: surprise data unraveled. Ph.D. thesis (2008)

[19] `https://github.com/hlt-bme-hu/hunspeech`
[20] `https://github.com/szekrenyesi/prosotool`

5. Hunyadi, L., Földesi, A., Szekrényes, I., Staudt, A., Kiss, H., Abuczki, A., Bódog, A.: Az ember–gép kommunikáció elméleti–technológiai modellje és nyelvtechnológiai vonatkozásai. In: Általános Nyelvészeti Tanulmányok XXIV: Nyelvtechnológiai kutatások, pp. 265–309. Akadémiai Kiadó, Budapest (2012)
6. Mertens, P.: The prosogram: Semi-automatic transcription of prosody based on a tonal perception model. In: Proceedings of Speech Prosody (2004)
7. Schmidt, T.: Good practices in the compilation of folk, the research and teaching corpus of spoken german. In: Kirk, J.M., Andersen, G. (eds.) Compilation, transcription, markup and annotation of spoken corpora, Special Issue of the International Journal of Corpus Linguistics [IJCL 21:3], pp. 396–418 (2016)
8. Szekrenyes, I.: Prosotool, a method for automatic annotation of fundamental frequency. In: 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom). pp. 291–296. IEEE, New York (2015)
9. Szekrényes, I.: Annotation and interpretation of prosodic data in the hucomtech corpus for multimodal user interfaces. Journal on Multimodal User Interfaces 8:(2), 143–150 (2014)
10. Szekrényes, I., Csipkés, L., Oravecz, C.: A hucomtech-korpusz és -adatbázis számítógépes feldolgozási lehetőségei, automatikus prozódiai annotáció. In: Tanács, A., Vincze, V. (eds.) VIII. Magyar Számítógépes Nyelvészeti Konferencia, pp. 190–198. JATEPress (2011)

III. Beszédtechnológia

Automatikus frázisdetektáló módszereken alapuló patológiás beszédelemzés magyar nyelven

Tündik Máté Ákos[1], Kiss Gábor[1], Sztahó Dávid[1], Szaszák György[1]

Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
e-mail:{tundik,kiss.gabor,sztaho,szaszak}@tmit.bme.hu

Kivonat A betegségek beszéd alapján történő korai diagnosztizálása során gyakori az automatikus osztályozási módszerek alkalmazása. Ezek az eljárások alkalmazhatók arra is, hogy Parkinson-kóros, valamint depressziós pácienseket az egészséges kontrollcsoport tagjaitól megkülönböztessük. A patológiás beszéd elemzése több szinten is elvégezhető; ebben a cikkben olyan prozódiai jellemzőket vizsgáltunk meg, melyek kinyerése automatikus hangsúly- és frázisdetektáló rendszerekből történt meg. Hipotéziseinket a frázisok időtartama és a frázisok szószámossága kapcsán fogalmaztuk meg. Az egészséges kontrollcsoport és a páciensek csoportja nagy mértékben elkülöníthető egymástól ezen paraméterek segítségével, melyeket statisztikai próbákból és SVM-alapú bináris osztályozásból származó eredményekkel alátámasztva az alábbi cikkben mutatunk be.
Kulcsszavak: gépi beszédfelismerés, gépi beszédértelmezés, hangsúly, frázis, Parkinson-kór, depresszió

1. Bevezetés

A betegségek beszéd alapján történő korai diagnosztizálása fontos kutatási terület, melyhez leginkább automatikus osztályozási módszereket alkalmaznak. Megvalósításukra számos példát találunk a szakirodalomban, pl. Bayes-típusú, SVM, mélyneurális háló, Random Forest, k-NN valamint GMM alapú [1] [2] [13]. A Parkinson-kór, valamint a depresszió esetén is cél, hogy a pácienseket az egészséges kontrollcsoport tagjaitól megkülönböztessük.

A Parkinson-kór az egyik leggyakoribb neurodegeneratív betegség, melynek a prevalenciája körülbelül 20/100 000 [7], előfordulása az életkor növekedésével egyenesen arányos. A betegség fő oka az agy feketeállományában lévő dopamintermelő idegsejtek nagymértékű sérülése, elhalálozása. A Parkinson-kór fő tünetei közé tartozik a remegés, az izommerevség és a kognitív károsodás. Egyéb tünetei közé tartozik pl. a depresszió, demencia, alvászavarok. Friss kutatások szerint lehetőség van a Parkinson-kór beszéd alapú detektálására [6] [8]. A legtöbb beteg esetén beszédzavarok is fellépnek(diszfónia, dizartria), valamint a beszéd minőségében is változás lép fel. Jellemzővé válik a csökkent hangerő, a fokozott hangremegés és a levegősség [3] [8].

Hasonlóképpen, lehetőség van a depresszió beszéd alapú vizsgálatára is. Ismertetőjelei közé tartozik a lassú beszédtempó és a monoton beszéddallam. Ezek a jellemzők konkrét akusztikai paraméterekhez köthetők, melyek szakirodalmi példákkal alátámaszthatóak. A hangulati, érzelmi állapotingadozás prozódiai paraméterekkel kapcsolható össze, mint pl. a ritmus, hanglejtés, hangsúly és az időzítés [15]. További vizsgált jellemzők lehetnek pl. az alapfrekvencia, a formánsok, a spektrális teljesítménysűrűség, kepsztrális vagy MFC együttható is [11] [12].

A [13] cikk szerzői a jitter, shimmer, HNR, alaphang jellemzőket a diszfónia különböző osztályainak megkülönböztetésére használták, majd ezeket összefüggésbe hozták a Parkinson-kór súlyosságának megállapításához használt UPDRS-értékekkel. Az osztályozás történhet kitartott hangok, olvasott szöveg, és spontán beszédfelvételek alapján. Az említett [13] cikkben 97%-os osztályozási teljesítményről számoltak be; Parkinson-kóros és egészséges csoportba történő bináris osztályzást végeztek, kitartott magánhangzók alapján. Egy másik cikk 85%-os pontosságról számol be; a beszéd érthetőségét folyamatos szöveggel vizsgálták, és "egészséges-enyhe-súlyos" Parkinson-kór osztályokat különböztettek meg [4].

A BME-TMIT Beszédakusztikai Laboratóriumában mind a Parkinson-kór, mind a depresszió beszéd alapú automatikus detektálása napjainkban is aktív kutatási terület [5] [10]. Jelen cikkben olyan prozódiai vonatkozású beszédjellemzőkre koncentrálunk, melyek automatikus hangsúly- és frázisdetektáló rendszerekből származnak. Feltételezésünk szerint az egészséges kontrollcsoport és a páciensek csoportja jól elkülöníthető egymástól ezen jellemzők segítségével, melyhez kapcsolódó kutatási eredményeinket az alábbi cikkben mutatjuk be.

A cikkünk az alábbi struktúra szerint épül fel: elsőként bemutatjuk a korpuszt, illetve a felhasznált automatikus hangsúly- és frázisdetektáló rendszereket. Ezután sor kerül a megkülönböztetési vizsgálatokhoz használt szempontrendszer leírására, végül ismertetjük az eredményeket.

2. Anyag és módszer

2.1. A felhasznált adatbázis

A kutatáshoz magyar nyelvű olvasott szöveget használtunk. Az egészséges kontrollcsoport résztvevői, valamint a depressziós és a Parkinson-kóros páciensek az "Az Északi szél és a Nap" c. történetet olvasták fel. Az említett történet egy fonetikailag kiegyensúlyozott rövid népmese (kb. hat mondat hosszú, és átlagosan 45 másodperc), gyakorlati jelentősége a foniátriai alkalmazásoknál van. Mivel a Parkinson-kóros és a depressziós páciensek különböző szövegezésű mesefordítást olvastak fel, ezért külön-külön egészséges kontrollcsoport összeállítására volt szükség annak érdekében, hogy a szövegezésből eredő különbségek méréseinket ne befolyásolják (a szöveg, illetve annak tagolása ugyanis nyilvánvalóan befolyásolja a prozódiát). A 36 fős Parkinson-os csoporthoz 32, az 52 fős depressziós csoporthoz 36 fős egészséges csoport tartozik. Mind a kontrollcsoport, mind a páciensek felvételei csendes környezetben készültek el.

A kísérleteinkhez szó-, szótag- valamint fonéma szintű szegmentálások álltak rendelkezésre, ezeket kényszerített illesztéssel készítettük el, utólagos kézi korrekcióval. Ezen szegmentálások alapján lehetséges a fonológiai frázisok automatikus detektálása, melyet a 2.2 és 2.3 fejezetekben ismertetünk. A fonológiai frázisok prozódiai egységet képeznek, saját hangsúllyal és intonációs görbével jellemezhetők. A prozódiai hierarchiában elfoglalt pozíciójuk szerint több fonológiai frázis egy intonációs frázist alkot. A magyar nyelvre a kötött hangsúlyozás jellemző - a szavak első szótagján -, és a fonológiai frázis - definíció szerint - pontosan egy hangsúlyos elemet tartalmaz. A kézi fonológiai frázisszegmentálások elkészítésétől eltekintettünk, csak az automatikus módszerek segítségével kinyert frázisok alapján végeztük el a kiértékelést.

2.2. GMM/HMM alapú automatikus frázisdetektáló rendszer

Ebben a fejezetben egy GMM/HMM alapú automatikus frázisdetektáló rendszert mutatunk be. A módszer hét különböző fonológiai frázis modellezésére és detektálására alkalmas, gépi tanulás segítségével. A felügyelt tanulás miatt címkézett tanítóanyag szükséges. A tanításhoz 11 állapotú HMM/GMM modelleket használtak, az akusztikai-prozódiai jellemzők közül az alapfrekvenciára és a széles sávú energiára volt szükség. A modellek tanulásánál az akusztikai jellemzők első és másodrendű deriváltjait is felhasználták.

A megnyilatkozást tartalmazó beszédfelvétel fonológiai frázisokra történő szegmentálása a Viterbi-algoritmus segítségével történik meg. Mivel a magyar hangsúlyozási szabályok szerint a hangsúly az első szótagra esik, a fonológiai frázisok és a hangsúly detektálása közel egyszerre történik meg. A Viterbi-algoritmus mindegyik fonológiai frázis előfordulásához azonos valószínűséget rendel. A fonológiai frázisok, frázisintervallumok sűrűségét külön paraméterrel kontrollálhatjuk, melynek következménye pl. a megnövekvő hamis-pozitív elemek száma. Bővebb leírás a módszerről az alábbi [9] cikkben található.

2.3. WCAD intonációs modell alapú automatikus frázisdetektáló rendszer

Ebben a fejezetben egy intonációs modellezési technikán alapuló automatikus frázisdetektáló rendszert mutatunk be. A WCAD (Weighted Correlation based Atom Decomposition) rövidítés magyar megfelelője a súlyozott korreláción alapuló atom dekompozíciós algoritmus, mely fiziológiai alapokon nyugszik (hangszalagokat mozgató izmok feszítettsége), nyelvileg releváns információt hordozva. Az intonációs görbe rekonstruálásának elméleti háttere a Fujisaki-modellből ered, mely a végleges kontúrt egy alap frekvenciakomponens, egy globális frázis "atom" valamint lokális "atomok" szuperpozíciójaként értelmezi.

A kísérleteinkhez kulcsfontosságú a lokális atomok időpontjának és amplitúdójának kinyerése. Az atomokat "Csúcs" és "Völgy" csoportra osztjuk, attól függően, hogy az adott szótagon jeleznek hangsúlyt, vagy a rákövetkezőn. A hangsúlyos szótagok kinyerése után a fonológiai frázisszegmentálás a magyar

nyelvre érvényes hangsúlyozási szabályok segítségével elkészíthető. A frázis kezdetén mindig hangsúlyos szótag ("Csúccsal" címkézve) áll, a végén pedig vagy "Völggyel" címkézett áll, ha a megnyilatkozásnak is vége, vagy pedig "Csúcs"-os, ha főmondat végén emelkedés következik be az intonációs görbében. A 2.2. fejezetben szereplő algoritmustól eltérően, az intervallumok bejelölése közvetlenül a beszédjelből kinyert jellemzők segítségével történik meg. Az egyes felvételekhez tartozó lokális atomok száma paraméterrel kontrollálható, mely kihat a fonológiai frázisokat tartalmazó intervallumok sűrűségére. A módszerről és annak hatékonyságáról részletesen a [14] számol be.

3. Az eredmények ismertetése

3.1. Hipotézisek

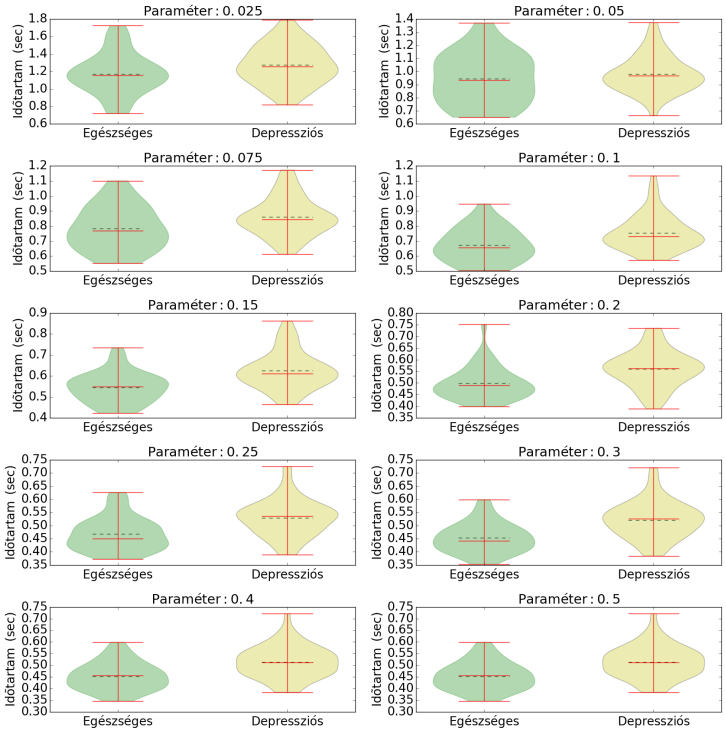
Korábbi vizsgálataink esetén egészséges emberek felvételeihez kézzel készített fonológiai frázis alapú szegmentációt használtunk fel, az automatikus módszerekkel történő összevetésre, az információelméletben használt fedés-pontosság (recall-precision) és F-mérték kiértékelési terében [14]. Ezen cikkben részletesen megvizsgáljuk az egészséges és beteg emberek közötti különbségeket a fonológiai frázisok aspektusában, de csakis az automatikus módszerek kimeneteire támaszkodva. Az alábbi szempontrendszert/hipotéziseket határozzuk meg:

- Frázisok időtartama - Azt feltételezzük, hogy az időtartam megkülönböztető jegy a betegek és az egészséges csoport között, mindkét frázisszegmentáló módszert alkalmazva.
- Frázisok szószámossága - Azt feltételezzük, hogy a frázisok hossza a benne lévők szavak számát tekintve megkülönböztető jegy a betegek és az egészséges csoport között, mindkét frázisszegmentáló módszert alkalmazva.
- t-próba és automatikus osztályozás az előző hipotézisekhez kapcsolódó eredményekre támaszkodva - Azt feltételezzük, hogy a frázisok időtartamának és szószámosságának megkülönböztető szerepe statisztikai úton is kimutatható, így ezek a jellemzők bináris osztályozásra is hatékonyan alkalmazhatók.

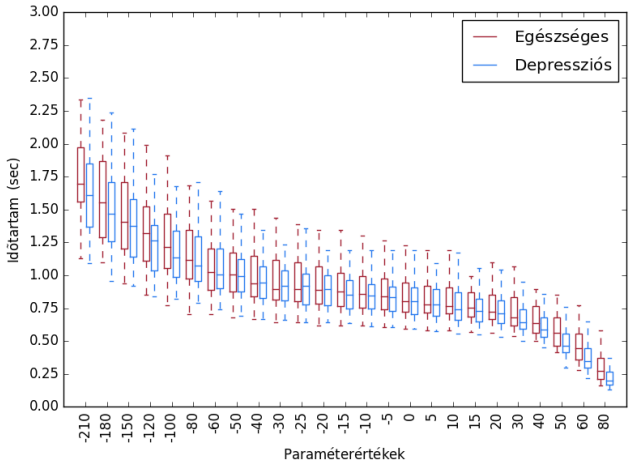
3.2. Frázisidőtartam elemzése

Az egyes frázisok időtartamának kinyerése a fonológiai frázisszegmentálás utófeldolgozó lépéseként könnyen elérhető. Ezeket mindkét módszer esetén kigyűjtöttük (a szünettel jelölt szakaszok kivételével), minden mérési szint esetén, melyek az algoritmusok paraméterkonfigurációiból adódnak. Az eredményeket Violin Plot-okon illetve egyszerű Box Plot-okon foglaltuk össze.

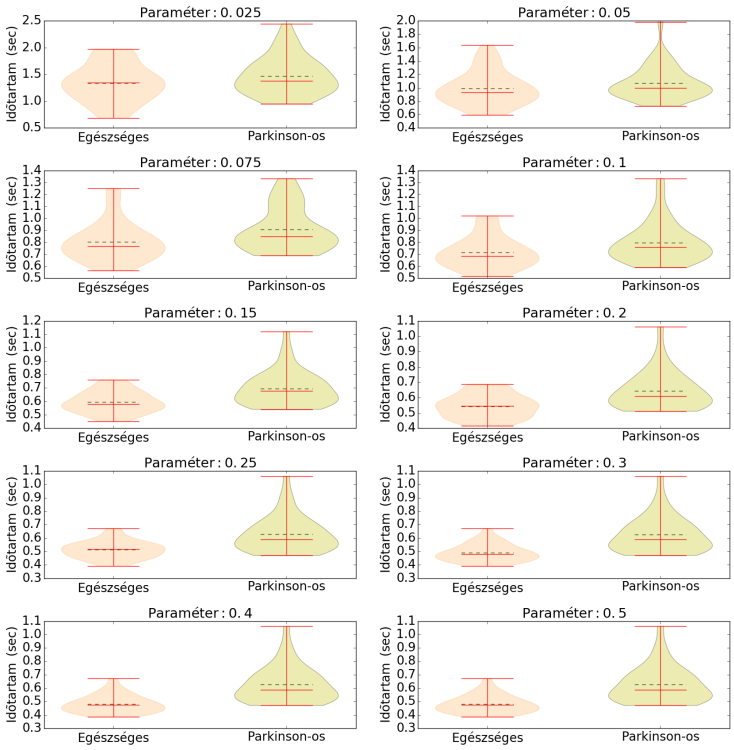
Az első megfigyelésünk az 1. és a 3. ábra alapján, hogy a frázisok időtartama egyre rövidebb lesz a WCAD-algoritmushoz tartozó paramétert magasabb értékre állítva (több frázishatároló kerül beszúrásra), valamint az egészséges csoport frázisai minden esetben rövidebbek, mint a Parkinson-os és depressziós csoporté.



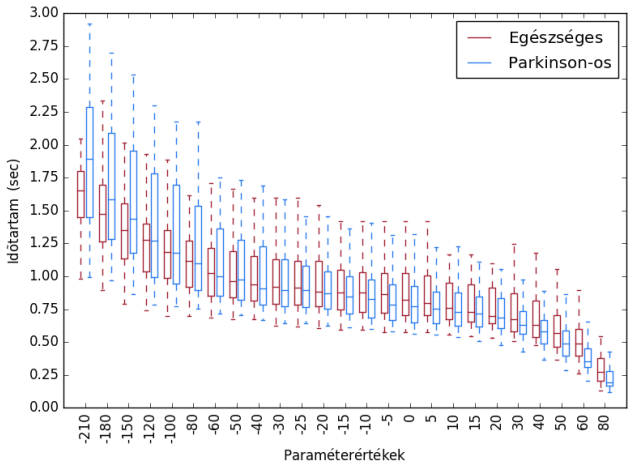
1. ábra. Frázisidőtartamok Violin Plot-ja, depressziós és egészséges csoportokra, WCAD-alapú automatikus frázisszegmentáló módszerrel



2. ábra. Frázisidőtartamok Box Plot-ja, depressziós és egészséges csoportokra, HMM-alapú automatikus frázisszegmentáló módszerrel



3. ábra. Frázisidőtartamok Violin Plot-ja, Parkinson-kóros és egészséges csoportokra, WCAD-alapú automatikus fráziszegmentáló módszerrel

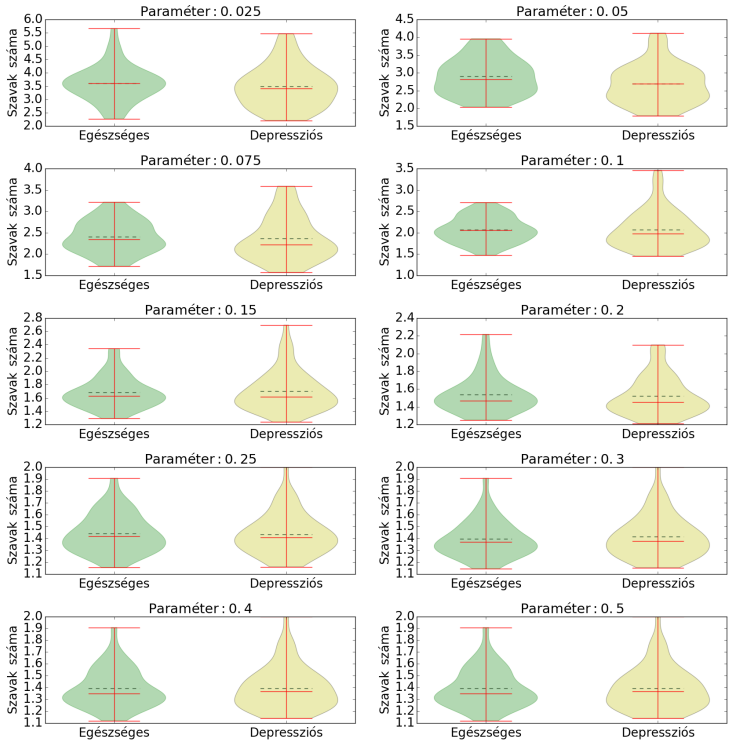


4. ábra. Frázisidőtartamok Box Plot-ja, Parkinson-kóros és egészséges csoportokra, HMM-alapú automatikus szegmentáló módszerrel

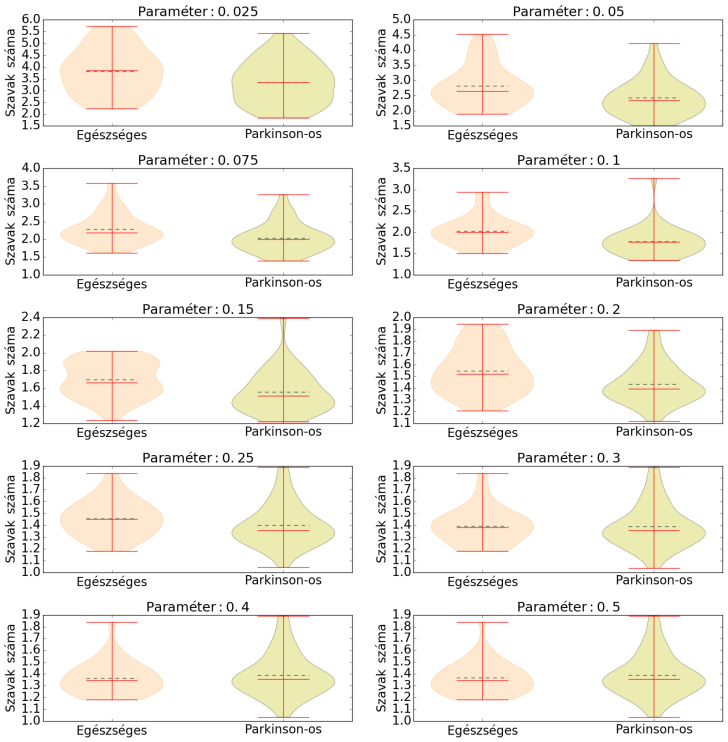
A következő megfigyelés a 2. és a 4. ábra alapján, hogy a frázisok időtartama a HMM-módszerhez tartozó "insertion log-likelihood" paraméter növelésével is csökken. Fontos különbség az előző módszer eredményéhez képest, hogy az egészséges csoportnál csak a mérési szintek első harmadában rövidebbek a frázisok, mint a Parkinson-osoknál, a depressziós csoporttal összehasonlítva pedig legtöbb esetben megegyeznek az értékek.

3.3. Frázisok szószámossága

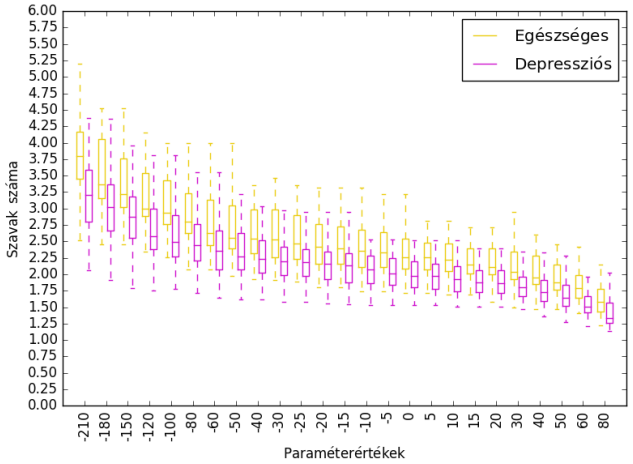
A frázisok szószintű kinyerése szintén utófeldolgozási feladat volt. A WCAD-alapú módszer során felhasználjuk a szótagok konkrét időpontját is. Így az egyes szavak időbeli illesztése az egyes frázishatárok közé egyszerűbb feladat, hiszen a frázis mindenképpen egy szó első szótagjával indul, valamint egy szó utolsó szótagjával végződik.



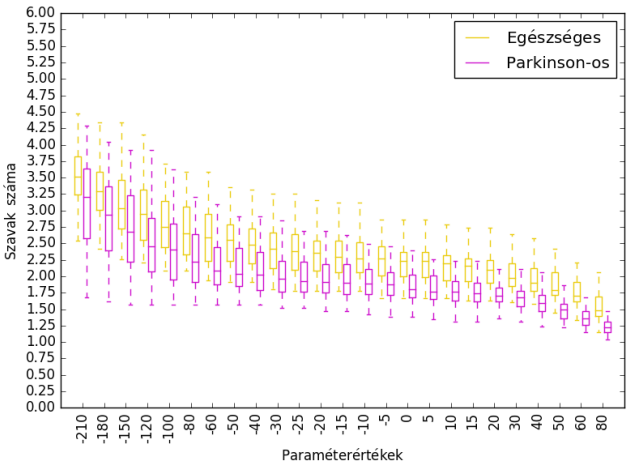
5. ábra. Frázisok átlagos szószámosságának Violin Plot-ja, depressziós és egészséges csoportokra, WCAD-alapú automatikus frázisszegmentáló módszerrel



6. ábra. Frázisok átlagos szószámosságának Violin Plot-ja, Parkinson-kóros és egészséges csoportokra, WCAD-alapú automatikus frázisszegmentáló módszerrel



7. ábra. Frázisok átlagos szószámosságának Box Plot-ja, depressziós és egészséges csoportokra, HMM-alapú automatikus frázisszegmentáló módszerrel



8. ábra. Frázisok átlagos szószámosságának Box Plot-ja, Parkinson-kóros és egészséges csoportokra, HMM-alapú automatikus frázisszegmentáló módszerrel

Más a helyzet a HMM-alapú rendszer esetén, ugyanis a rendszer nem használ szótag-információt, ezért a szavak frázisokra történő időbeli illesztésénél figyelnünk kell az időben átfedő szakaszokra is. Ezenkívül egyes frázisok közepén szünet is szerepel, mely pl. kézi szegmentációval nem fordulhatna elő; ilyenkor a szünetek mentén szét kell bontani a frázist, hogy a prozódiai hierarchiának megfelelő tagolást kapjunk.

A fent leírtak után elvégeztük a frázisok szószámosságának megállapítását, összesítve az egyes szakaszokban lévő szavakat, mindkét metódus mindegyik mérési szintjére, melyeket Violin Plot-okkal és Box Plot-okkal ábrázoltunk ismét.

Az első megfigyelésünk az 5. és a 6. ábrákkal kapcsolatban, hogy az egyes frázisokra kevesebb szó jut, a WCAD-alapú algoritmus paraméterét növelve(ez összefüggésben van az időbeli jellemzőknél leírtakkal). Az egészséges csoport frázisszintű megnyilatkozásai csak a mérési pontok első felében rövidebbek, mint a Parkinson-os csoporté, a depressziós csoporttal összevetve pedig közel azonosak.

Hasonlóképpen a 7. és a 8. ábrákon látható, hogy a HMM-alapú módszer paraméterét növelve a frázishosszok rövidebbek lesznek általánosságban, viszont a megfigyelések különbözőek; az egészséges csoportot összevetve a Parkinson-os és a Depressziós csoporttal minden mérési szint esetén rövidebb frázishosszokkal találkozunk.

3.4. t-próba és automatikus osztályozás

Az előző alfejezetben leíró jellegű statisztikai eredményekről számoltunk be, melyek a frázisok időtartamához és a bennük foglalt szavak számosságához kapcsolódtak. Következő lépésben a t-próba elvégzésével megvizsgáltuk, hogy a jellemzők esetén megfigyelt különbségek szignifikánsak-e. A frázisok időtartamát és

szószámosságát a frázisszegmentáló módszerek különböző paraméterértékeinek beállításával vizsgáltuk. WCAD alapú esetben 10, HMM alapú esetben pedig 25 különböző értéket használtunk. A H_0 hipotézis szerint az adott jellemzőnek nincs megkülönböztető szerepe az aktuálisan vizsgált módszer paraméterbeállítása mellett, ellenkező esetben a H_1 teljesül.

1. táblázat. Megkülönböztetés frázisidőtartam és a frázisokban lévő átlagos szószámosság alapján, mindkét frázisszegmentáló módszerrel

	t-próba eredmények($\alpha = 0.05$)		
	H_0	H_1	H1-paraméterek
Frázisidőtartam, Parkinson, WCAD	1	9	0.05—0.5
Frázisidőtartam, Depresszió, WCAD	1	9	0.025, 0.075—0.5
Frázisok szószámossága, Parkinson, WCAD	5	5	0.05—0.2
Frázisok szószámossága, Depresszió, WCAD	10	0	—
Frázisidőtartam, Parkinson, HMM	22	3	50—80
Frázisidőtartam, Depresszió, HMM	22	3	50—80
Frázisok szószámossága, Parkinson, HMM	0	25	-210—80
Frázisok szószámossága, Depresszió, HMM	0	25	-210—80

Az 1. táblázatból leolvasható, hogy a WCAD-alapú algoritmus az egészséges kontrollcsoportot és a pácienseket a frázisok időtartamát tekintve képes megkülönböztetni. Ezen kívül a Parkinson-os csoporttal összevetve a frázisokban szereplő szavak száma is megkülönböztető erővel bír, habár ez csak a mérési szintek felében igaz. Összehasonlításképpen, a HMM-alapú megoldás megkülönböztető ereje a frázisokban lévő szavak számában rejlik.

A következőkben az osztályzási kísérleteket és azok eredményeit ismertetjük. Minden kísérlet esetén az SVM C-SVC típusát használtuk, lineáris kernellel. A C hiperparaméter kimerítő kereséssel lett megállapítva, 2 első tíz hatványának szisztematikus végigpróbálgatásával. A tanítás és a tesztelés során „leave-one-out" keresztvalidációt alkalmaztunk.

2. táblázat. Eredmények SVM-alapú osztályzással

	SVM-eredmények	
	Pontosság (Acc)	C
Parkinson, WCAD	70,6%	8
Depresszió, WCAD	67 %	2
Parkinson, HMM	79,4%	2
Depresszió, HMM	62,5%	4
Parkinson, HMM+WCAD	88%	1
Depresszió, HMM+WCAD	81%	16
Parkinson, HMM+WCAD+FFS	**91%**	**1**
Depresszió, HMM+WCAD+FFS	**83 %**	**16**

Az eredmények a 2. táblázatban láthatók. Ha az egyes módszerekből származó jellemzőket külön tekintjük, a bináris osztályzás a Parkinson-Egészséges kontrollcsoport esetben a legpontosabb, a HMM-alapú fonológiai frázisszegmentálás jellemzőivel. Ha a két módszerből származó jellemzőket egyesítjük, további jelentős javulást érünk el, mind a "depressziós", mind a "Parkinson-os" esetben.

Végül, a Fast Forward Selection (FFS)-alapú jellemzőkiválasztással kaptuk a legjobb eredményeket, amelynek során minden egyes lépésben azt a jellemzőt választjuk ki az összességből, amellyel a legnagyobb mértékű javulás következik be a pontosság (accuracy) értékében. Depresszió esetén közel 40%-kal csökkent a dimenziók száma (37-ről 22-re), az eltávolított paraméterek főként a HMM-es algoritmusból származtak. Parkinson-os esetben még ennél is nagyobb, közel 90%-os dimenziócsökkenés következett be (42-ről 5-re), a megmaradt jellemzők többsége a WCAD-algoritmusból származik.

4. Összegzés

Cikkünkben részletesen bemutattuk az egészséges és Parkinson-kóros, valamint depressziós páciensek közötti különbségeket a fonológiai frázisok aspektusában, automatikus módszerek kimeneteire támaszkodva. Mivel a fonológiai frázisok egyszerre jellemzik a hangsúlyozást, az intonációt, illetve a prozódiai tagolást, lényegében ezen jellemzők tekintetében informatívak a kapott eredmények. Hipotéziseinket statisztikai vizsgálatokkal, valamint automatikus osztályozási kísérletekkel támasztottuk alá. Beigazolódott, hogy az automatikusan kinyert frázisok időtartama, valamint a frázisokra eső szószámosság megkülönböztető jegyek a betegek és az egészséges csoport között.

Az eredményeket értelmezve Parkinson-kóros felvételeknél a megakadások, szókeresések miatt kevesebb szó alkot egy frázist, a frázisok azonban időben így is hosszabbak, mint az egészségeseknél. Depressziósok esetén a frázisok nyúlnak, de kb. azonos szószámosságúak, mint az egészségeseknél. Néhány realizáció összehasonlítása alapján feltételezzük, hogy a monotonitás ellenére a frázishatárokat a rendszer többnyire ugyanott detektálja (nem lesz több szó a frázisban átlagosan), viszont a kisebb hangsúlyú, lapos intonációjú frázistípusok részaránya megnövekszik. Ezt a feltételezést a jövőben további kísérletekkel tervezzük igazolni. A legcélravezetőbb a HMM- és WCAD-jellemzők kombinált alkalmazása volt. A Parkinson-kór elkülönítése pontosabb a prozódia alapján (91%), ugyanakkor a depresszió esetén kapott 83% is kielégítőnek tekinthető.

További terveink között szerepel a magyar nyelvű adatbázis bővítése (a kísérletek elvégzése nagyobb elemszámmal), valamint egyéb jellemzőkkel történő együttes osztályzás és idegen nyelvű felvételek vizsgálata.

5. Köszönetnyilvánítás

A szerzők köszönetüket fejezik ki a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatalnak, amely a PD-112598 projekt keretében a kutatást támogatta.

Hivatkozások

1. Erdogdu Sakar, B., Isenkul, M., Sakar, C.O., Sertbas, A., Gurgen, F., Delil, S., Apaydin, H., Kursun, O.: Collection and Analysis of a Parkinson Speech Dataset with Multiple Types of Sound Recordings. in IEEE Journal of Biomedical and Health Informatics, vol. 17, no. 4, 828-834, (2013).
2. Frid, A., Safra, E.J., Hazan, H., Lokey, L.L., Hilu, D., Manevitz, L., Ramig, L.O.; Sapir, S.: Computational Diagnosis of Parkinson's Disease Directly from Natural Speech Using Machine Learning Techniques. in Proc. of the 2014 IEEE International Conference on Software Science, Technology and Engineering, 50-53, (2014).
3. Harel, B., Cannizzaro, M., Snyder, P. J.: Variability in fundamental frequency during speech in prodromal and incipient Parkinson's disease: A longitudinal case study. in Brain and Cognition, vol. 56, 24-29, (2004).
4. Khan, T., Westin, J., Dougherty, M.: Classification of speech intelligibility in Parkinson's disease. in Biocybernetics and Biomedical Engineering, vol. 34, Issue 1, 35-45, (2014).
5. Kiss, G., Tulics, M. G., Sztahó, D., Esposito, A., Vicsi, K.: Language Independent Detection Possibilities of Depression by speech. in Recent Advances in Nonlinear Speech Processing, 103-114, (2016).
6. Little, M. A., McSharry, P. E., Hunter, E. J., Spielman, J., Ramig, L. O.: Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. in IEEE Transactions On Biomedical Engineering, vol. 56, no. 4, 1015-1022, (2009).
7. Rajput, M., Rajput A., Rajput, A. H.: Epidemiology. in Handbook of Parkinson's disease. R. Pahwa and K. E. Lyons, Eds., 4th ed., Informa Healthcare, (2007).
8. Sapir, S., Ramig, L., Spielman, J., Fox, C.: Formant Centralization Ratio (FCR): A proposal for a new acoustic measure of dysarthic speech. in Journal of Speech, Language and Hearing Research, vol. 54, 114-125, (2010).
9. Szaszák, G., Beke, A.: Exploiting Prosody for Automatic Syntactic Phrase Boundary Detection in Speech. in Journal of Language Modelling, vol. 0, no. 1, 143–172, (2012).
10. Sztahó, D., Vicsi, K.: Estimating the Severity of Parkinson's Disease Using Voiced Ratio and Nonlinear Parameters. in Proc. of 4th International Conference on Statistical Language and Speech Processing, Pilsen, Czech Republic, 96-107, (2016).
11. Terapong B. at all: Assessment of Vocal Correlates of Clinical Depression in Female Subjects with Probabilistic Mixture Modeling of Speech Cepstrum. 11th International Conference on Control, Automation and Systems, (2011).
12. Thaweesak Y. at all: Characterizing Sub-Band Spectral Entropy Based Acoustics as Assessment of Vocal Correlate of Depression. International Conference on Control, Automation and Systems, (2010).
13. Tsanas, A., Little, M.A., McSharry, P.E., Spielman, J., Ramig, L.O.: Novel Speech Signal Processing Algorithms for High-Accuracy Classification of Parkinson's Disease. in IEEE Trans. on Biomedical Engineering, vol.59, no.5, 1264-1271, (2012).
14. Szaszák, G., Tündik, M. A., Gerazov, B., Gjoreski, A.: Combining Atom Decomposition of the F0 Track and HMM-based Phonological Phrase Modelling for Robust Stress Detection in Speech. in Proc. of 18th International Conference on Speech and Computer, Budapest, Hungary, 165-173, (2016).
15. Vicsi, K., Sztahó, D.: Problems of the Automatic Emotion Recognitions in Spontaneous Speech; An Example for the Recognition in a Dispatcher Center. in Esposito, A. et al.: Toward Autonomous, Adaptive, and Context-Aware Multimodal Interfaces. Theoretical and Practical Issues. Heidelberg: Springer, 331-339, (2011).

Depresszió súlyosságának becslése beszédjel alapján magyar nyelven

Gábor Kiss[1], Lajos Simon[2], Klára Vicsi[1]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
{kiss.gabor, vicsi}@tmit.bme.hu
[2]Semmelweis Orvostudományi Egyetem Pszichiátriai és Pszichoterápiás Klinika,
simon.lajos@med.semmelweis-univ.hu

Kivonat: A depresszió korunk egyik legelterjedtebb, gyógyítható betegsége, ám diagnosztizálása szaktudást igényel, és így a kórkép felállítása a társadalom egy szűk rétegére hárul. A depresszió súlyossága nagyban befolyásolja az ebben szenvedő beteg életminőséget. Depresszió hatására megváltoznak az emberi beszédproduktum egyes jellemzői, amelyek számszerűsíthetőek és mérhetőek. Emiatt lehetőség nyílik a depresszió beszédjel alapú detektálásra, ami megkönnyítheti, illetve szélesebb körben lehetővé teheti a betegség diagnosztizálását. Ezen okok miatt fontos kutatási terület a depressziós állapot beszédjel alapú felismerése és súlyosságának becslése. Ebben a cikkben bemutatunk egy Szupport Vektor Regressziós számításon alapuló automatikus rendszert, ami képes a beszédjel alapján megbecsülni nemcsak a depresszió meglétét, hanem a beszélő állapotának súlyosságát is. Megvizsgáljuk, hogyan változik a rendszer pontossága, ha külön rendszert alkalmazunk a nők és a férfiak esetén, illetve ha felhasználjuk a beszéd fonéma szintű szegmentálását a beszédet leíró jellemzők előállítása során.

Kulcsszavak: depresszió, beszédjel alapú detektálás, beszédszegmentálás, regresszió, SVR

1 Bevezetés

Az emberi beszédproduktum sokrétű jelentést hordoz, így nem csupán a beszéd nyelvi tartalmát közvetíti, hanem számos, a kommunikációval kapcsolatos nonverbális üzenetet is hordoz, mint például a beszélő érzelmi töltete, és mindezek mellett a beszélőre jellemző fiziológiai állapotot is tükrözi. Pszichiátriai szakorvosok állítják, hogy a páciens beszéde alapján képesek annak pszichofiziológiai állapotát felmérni, így például a depressziót is. A depressziós betegek beszédét a szakorvosok a következő jellemzőkkel szokták leírni: fakó, monoton, élettelen. Természetesen ezek az érzeti jellemzők számszerűsíthetőek, és így kapcsolatba hozhatók a beszéd egyes akusztikai és fonetikai jellemzőivel, mint például alapfrekvencia, formáns frekvenciák, beszédtempó stb. Ezt a jelenséget már 1921-ben megfigyelte és publikálta Emil Kraepelin, a modern pszichiátria egyik megalapozója [10].

A WHO (World Health Organization) 350 millióra becsülte a depresszióban szenvedő betegek számát 2012-ben [11]. A WHO előrejelzései szerint 2030-ra a depresszió

a három legsúlyosabb betegség között lesz világviszonylatban a HIV/AIDS vírus és a szívbetegségek mellett [12]. Annak ellenére, hogy a betegségben szenvedők száma igen magas, a diagnózis felállítása egy kisszámú képzett szakorvosrétegre hárul. A depreszsziós betegek életminőségé a depresszió következtében és hatására erősen romlik, a tünetek súlyosságától függően akár képtelenek rendszeresen dolgozni, ami komoly gazdasági problémát jelent a társadalomnak. Ráadásul a súlyos depresszió megnöveli az öngyilkossági kockázatot is [3].

Ezek miatt hasznos lenne egy olyan objektív, robosztus diagnosztizáló rendszer kialakítása, amelyet akár nem szakképzett orvosok is használhatnának a felismerésére és követésére.

A depresszió és a beszéd kapcsolata már az 1980-as évektől kezdve fontos kutatási területnek számít, és több akusztikai illetve fonetikai paramétert kapcsolatba hoztak a depresszióval, mint például az átlagos alapfrekvencia értéket, az alapfrekvencia tartományát, beszédtempót [15]. Azonban a depresszió gépi detektálása új kutatási területnek számít, amit az informatika fejlődése tett lehetővé. Cummins és társai 2015-ben a Speech Communication folyóiratban közöltek egy átfogó tanulmányt a beszédjelalapú depresszió detektálásához kapcsolódó fontosabb kutatások legfrissebb eredményeiről [6].

Alapvetően kétféle gépi detektálási módszert alkalmaznak a kutatók: osztályozó eljárást, amely a beszélő depressziós állapotát detektálja, illette regressziós eljárást, amely megbecsüli a depresszió súlyosságát. Ami közös bennük, hogy mindkét eljáráshoz szükség van valamilyen orvosi besorolási rendszerre. A két legelterjedtebb besorolási rendszer a Hamilton Rating Scale for Depression (HAMD) [7] és a Beck Depression Index (BDI) [1]. Mi ebben a cikkben a BDI továbbfejlesztett változatát használjuk, a BDI-II skálát [1].

A depressziós állapot az agy motorikus működését befolyásolja, emiatt változik a depressziós ember beszédproduktuma. Elsősorban természetesen az a kérdés, hogy a depresszió hatására mely akusztikai és fonetikai jellemzők változnak meg. A nemzetközi irodalom több beszédparamétert is említ, ami a depresszió hatására megváltozik. Azonban abban még nincs általános egyezség, hogy az adott beszédparamétereket hogyan érdemes mérni. Természetesen a mérési lehetőség nagyban függ a beszédadatbázis feldolgozottságától, ami lehet csupán egyszerű beszéd/nembeszéd szerinti, de akár pontos fonéma színtű szegmentálása is az adatbázisnak. Az utóbbit értelemszerűen lényegesen költségesebb megvalósítani, emiatt a legtöbb eddigi kutatásban nem alkalmaztak fonéma színtű szegmentálást, viszont ebből kifolyólag egyes beszédparamétereket csak pontatlanul, nagyobb szórással tudtak megmérni, illetve egyes beszédparaméterek meg sem mérhetőek a beszéd szegmentálása nélkül. Talán pont ezen okokból kifolyólag depresszió esetén a kutatók az egyes beszédparamétereknél eltérő tendenciákat mértek. [13][14]. Egy másik alapvető eltérés a különböző kutatásokban, hogy a női és a férfi beszédmintákat hol egyben kezelik, hol pedig külön. Ez részben függhet attól is, hogy a jelenleg létező depressziós beszédadatbázisok viszonylag kisméretűek: 30-160 beszélőtől tartalmaznak beszédmintákat [6], így ha külön vizsgálják a nőket és a férfiakat, akkor a vizsgált halmaz mérete tovább csökken. Ugyanakkor a beszélő neme nagyban befolyásolhatja az egyes beszédjellemzők értékeit, így ha közös modellt használunk a nők és a férfiak esetében, az problémát okozhat a depressziós állapot felismerésében, ami mindenképpen megoldandó feladat. Másrészről szakorvosok állítják, hogy a férfiak és a nők beszédében nem ugyanúgy realizálódik minden esetben a depressziós

állapot, így ez kifejezetten indokolhatja a beszélő neme szerinti eltérő eljárás kidolgozását.

Ebben a cikkben egy olyan gépi eljárást mutatunk be, ami képes a depresszió súlyosságának becslésére a vizsgált személy beszédjele alapján magyar nyelv esetén. Továbbiakban bemutatjuk, hogy mennyiben változik az eljárás pontossága, ha külön kezeljük a férfiakat és a nőket, illetve ha alkalmazunk automatikus beszédszegmentálót, így lehetőségünk nyílik egy másféle jellemzőhalmaz kinyerésére. A következő két hipotézist állítottuk fel a munka megkezdése előtt: A depresszió súlyossága pontosabban becsülhető, ha külön modellt használunk a férfiak és a nők esetén (H1), illetve a depresszió súlyossága pontosabban becsülhető, ha a jellemzők kiszámításhoz felhasználjuk a beszéd fonéma szintű szegmentálását (H2).

A cikk a következő felépítést követi. A bevezetés után a második fejezetben bemutatjuk a használt beszédadatbázist. A harmadik fejezetben a munka során használt módszereket. A negyedik fejezetben tárgyaljuk az elvégzett kísérleteket. Az ötödik fejezetben az eredmények értékelése történik meg. Majd a hatodik fejezetben összefoglaljuk a munka eredményeit.

2 Adatbázis

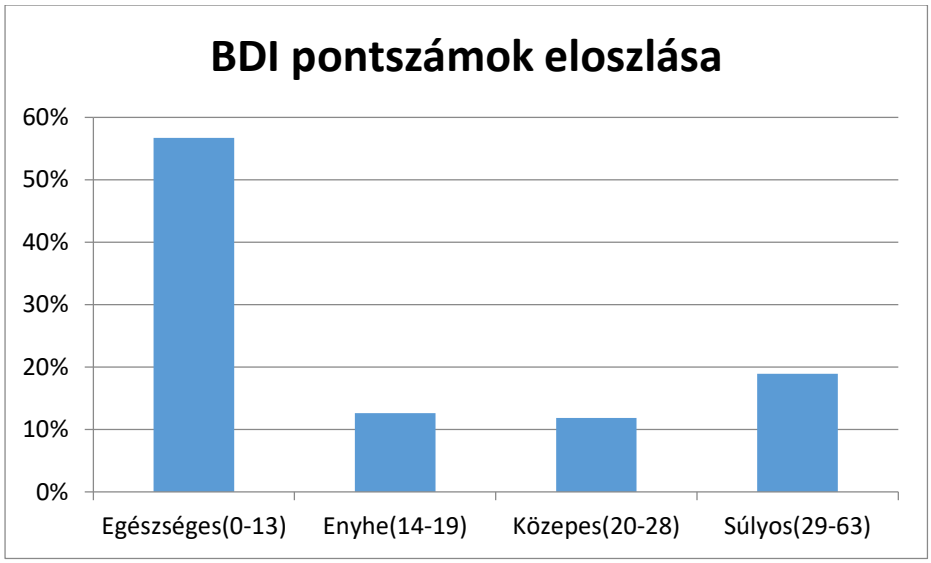
A beszédminták gyűjtését a Semmelweis Egyetem Pszichiátriai és Pszichoterápiás Klinikájával együtt végeztük. A beszédminták gyűjtésénél törekedtünk arra, hogy a beszélők lefedjék a depresszió súlyosságának különböző fokozatait, az egészséges állapottól az egészen súlyos depresszióig. A vizsgált személyeknek egy fonetikusan kiegyensúlyozott mesét ("Az északi szél és a Nap") kellett felolvasniuk, ami széles körben elterjedt a miénkhez hasonló vizsgálatokban. A felvételek csendes helyiségben kerültek rögzítésre 44,1 kHz mintavételi frekvenciával. A cikk során erre a beszédadatbázisra „Magyar Depressziós Adatbázis”-ként hivatkozunk.

Az adatbázisba gyűjtött felvételekhez elkészítettük az egyes felvételekhez tartozó fonéma szintű szegmentálást, a labor által fejlesztett automatikus szegmentáló program segítségével [8]. Minden esetben rögzítésre került a BDI-II-es pontszám, amely az adott személy depressziójának súlyosságát írja le. A BDI-II skála 0-tól 63-ig terjed, ahol a 0 az egészséges állapotot jelöli, míg a 63 a legsúlyosabb depressziós állapotot. A BDI-II skála pontszámaihoz a következő besorolás adott: 0-13 egészséges, 14-19 enyhe depresszió, 20-28 közepes depresszió, 29-63 súlyos depresszió. A BDI pontszámok 0-tól 50-ig fordultak elő a gyűjtött mintákban. A vizsgált személyek átlagéletkora 42,2 év volt, (-+14,4; min.: 20; max.: 65). Az 1. táblázatban láthatóak az adatbázis főbb jellemzői.

1. Táblázat: A Magyar Depressziós Adatbázis főbb jellemzői.

Bemondók száma	BDI-értékek átlaga	BDI-értékek szórása
127 (nő:79/ffi:48)	14,2 (nő:14,7/ffi:13,3)	13,5 (nő:14/ffi:12,7)

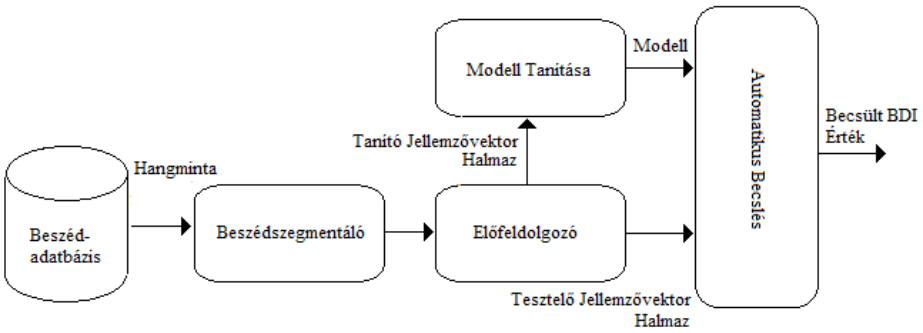
Az 1. ábra mutatja BDI pontszámok eloszlását az adatbázisban szereplő személyeknél, a depresszió súlyossága alapján.



1. ábra: A vizsgált személyek BDI-értékük szerinti eloszlása a Magyar Depressziós Adatbázisban.

3 Módszerek

A munka során használt automatikus gépi eljárás folyamatábráját a 2. ábrán lehet látni. Az eljárás először szétválasztja a beszédadatbázist tanító és tesztelő mintákra. Ezután a beszédmintákat szegmentálja és címkézi fonéma szinten. Majd az így felcímkézett hangmintákon elvégzi az akusztikai és fonetikai jellemzők kiszámítását. A kiszámított értékekből jellemzővektorokat generál, és az egyes paraméterek értékeit -1 és 1 közötti skálára normálja. Tanítás esetén ezekből a jellemzővektorokból készíti el a regressziós modellt adott gépi tanuló eljárással. A tesztelés során az eljárás az adott jellemzővektor és a regressziós modell alapján becsüli meg a vizsgált mintához tartozó beszélő depressziójának súlyosságát és rendel hozzá BDI pontszámot.



2. ábra: Az automatikus gépi eljárás folyamatábrája.

3.1 Előfeldolgozás

A beszédminták 16 kHz-en újra lettek mintavételezve és csúcsra normálva.

Korábbi tapasztalataink alapján és a nemzetközi irodalommal összhangban a következő jellemzőket használtuk a vizsgálatok során: alapfrekvencia, intenzitás, melsávos energiaértékek, jitter, shimmer, formánsértékek (első és második), formánsok sávszélességei (első és második). Ezen jellemzők 10 ms-os lépésközzel Praat program segítségével kerültek kiszámításra [2]. Majd a számított étékekből a következő statisztikai függvények használatával – átlag, korrigált szórás, percentilis tartomány (a rendezett mintahalmaz alsó-felső 2,5%-nak elhagyása után képzett különbsége a maximum és minimum értéknek) – nyertük ki a hangmintákhoz rendelt jellemzőket. Ezt a jellemzőhalmazt még tovább bővítettük az artikulációs sebesség, a beszédtempó és a tranziensarány (rate of transients) jellemzőkkel [9].

Az akusztikai és fonetikai jellemzőket két osztályba lehet sorolni, a szegmentális és a prozódiai jellemzőkre. A szegmentális jellemzők számításának a helye nagyban befolyásolja a statisztikai jellemzők számított értékét. Emiatt kétféleképpen számítottuk ki ezeket, az egyik esetben úgy, hogy a zöngés szakaszon mért értékből képeztük a statisztikai jellemzőket (ez a számítás elvégezhető szegmentálás nélkül), illetve úgy is, hogy az adott bemondás összes „e” hangjának a közepén mért értékekből lettek képezve a statisztikai jellemzők. Azért az „e” hangot választottuk, mert ez a magánhangzó fordult elő leggyakrabban a felolvasott mesében.

3.2 Regressziós gépi tanuló eljárás

A kísérletek során gépi tanuló eljárásnak a Support Vector Regression (SVR) eljárást választottuk [16], ami a Support Vector Machine (SVM) regressziós feladatokra alkalmas változata [5]. Az SVM az általános lineáris osztályozók családjába tartozik, ám képes nemlineáris problémák megoldására is a kernel függvény megfelelő megválasztásával. Az SVR(SVM) egyedi tulajdonsága, hogy egyidejűleg minimalizálja a regressziós eljárás hibáját, és közben maximalizálja az eljárás általánosító képességét.

A kísérletek megvalósítása során a LibSVM 3.20 verzió számú függvény könyvtárat használtuk [4], Radial Basis Function (RBF) kernellel, a kernel által használt hiperparamétereket (cost és gamma) Grid Search kereséssel optimalizáltuk úgy, hogy a 2 hatványai lettek kipróbálva -10 és +10 között.

3.3 Jellemzővektor kiválasztás

Az SVR pontosságát nagyban befolyásolja a megfelelő jellemzővektorok kialakítása, vagyis a lényegtelen, zajszerű jellemzők elhagyása. Ez hatványozottan igaz a kis számú tanító mintahalmaz esetén, mint például a mi esetünkben is. Többféle jellemző kiválasztási algoritmus létezik a jobban teljesítő jellemzővektorok előállítására, mi a Fast Froward Selection (FFS) eljárást használtuk. Ennek az eljárásnak a lényege, hogy viszonylag gyorsan kiválaszt egy, az algoritmus által optimálisnak ítélt n elemű jellemzőhalmazt, ami az algoritmus által előállított eltérő számú, szuboptimális jellemzőhalmazok közül a legjobban teljesít. Az eljárás az üres jellemzőhalmazból indul ki. Az *i*-

dik lépésben rendelkezésére áll az algoritmus szerinti legjobb *i-1* elemszámú jellemzőhalmaz, és ezt bővíti ki a legjobb *i* elemszámú jellemzőhalmazra úgy, hogy megvizsgálja, melyik eddig még fel nem használt jellemző hozzáadásával kapható a legjobb pontosan *i* elemszámú jellemzőhalmaz. Az előnye, hogy viszonylag gyors, a hátránya, hogy ha a *k*-dik lépésben beválaszt egy jellemzőt a jellemzőhalmazba, az onnantól kezdve minden halmazban benne lesz, ami *k* vagy annál nagyobb méretű.

3.4 A tesztelési eljárás

Az adatbázisban szereplő, viszonylag alacsony mintaszám miatt az ilyenkor szokásos, leave one out keresztvalidációs eljárást (LOOCV – leave one out cross validation) használtuk a tesztelések során minden esetben, így például az FFS alkalmazása és a hiperparaméterek optimalizálása során is. Az eljárás lényege, hogy a rendszer pontosságának a leírására mindegyik mintát pontosan egyszer felhasználja mint teszthalmaz, míg a maradék mintákat mint tanítóhalmazt, és így a tesztelőmintákon kapott eredmények írják le a rendszer teljes mintahalmazon számított pontosságát.

3.5 Az eljárás pontosságának mérése

Regressziós feladatok pontosságának jellemzésére többféle mérőszámot is lehet használni az adott módszer pontosságának leírására. Mi a következő három leíró jellemzőt választottuk, amelyek széles körben elterjedtek regressziós eljárások pontosságának leírására: az átlagos hibaértékét (MAE – mean absolute error), az átlagos négyzetes hibaértéknek a gyökét (RMSE – root mean square error), illetve az eredeti BDI pontszámoknak és az eljárás által becsült BDI pontszámoknak a Pearson-féle korrelációs együttható értékét.

4 Kísérletek

Összesen négy kísérletet hajtottunk végre a vizsgálat során. Minden kísérlet esetén külön alkalmaztuk az FFS eljárást, illetve optimalizáltuk a hiperparamétereket.

Az első kísérletnél együtt kezeltük a női és férfi mintákat, és csak olyan jellemzőket használtunk fel a jellemzővektorok kialakítása során, amelyeket a beszédjel szegmentálása nélkül is ki lehet számítani. Így a 3.1 alfejezetben tárgyalt jellemzők közül az artikulációstempó, a beszédtempó és az „e” hangokon számított szegmentális paraméterek statisztikai értékei nem kerültek bele ebbe a vizsgálatba. Erre a kísérletre a továbbiakban mint „*baseline*” kísérletre hivatkozunk. Azért neveztük el *baseline* kísérletnek, mivel a többi általunk elvégzett kísérlet ennek a „továbbfejlesztett” változata, ami speciálisabb előfeldolgozást illetve kialakítást igényelt.

A második kísérlet során ugyanazzal a jellemző halmazzal dolgoztunk, mint a *baseline* kísérlet esetében, de külön modellt hoztunk létre a női és férfi minták esetén. A gyakorlatban ez azt jelentette, hogy nemek szerint külön végeztük el a jellemzővektorok kialakítását (FFS), a hiperparaméter optimalizációt és a tesztelést. Erre a kísérletre

a továbbiakban, mint „*gender*” hivatkozunk. Ennek a lényege az volt, hogy megvizsgáljuk, hogyan módosul a *baseline* regressziós eljárás pontossága, ha nemek szerint eltérő modellt alkalmazunk, és ennek a kísérletnek a segítségével igazolhatjuk vagy cáfolhatjuk a H1 hipotézisünket.

A harmadik kísérletben hozzávettük azokat a jellemzőket is a vizsgálathoz, amelyek kiszámításához a beszédjel szegmentálása szükséges, vagyis a 3.1 alfejezetben felsorolt összes jellemzőből kerültek kialakításra a jellemzővektorok az FFS eljárás segítségével. De a női és a férfi mintákat együtt kezeltük, mint a *baseline* kísérlet esetében. Erre a kísérletre a továbbiakban mint „segemented” kísérlet hivatkozunk. Ennek a lényege az volt, hogy megvizsgáltuk, hogyan módosul a *baseline* regressziós eljárás pontossága, ha egyes jellemzőket nemcsak a beszéd egészén mérjük, hanem előre definiált pontos helyeken felhasználva a beszéd fonéma szintű szegmentálását, illetve a szegmentálás felhasználása által képesek voltunk artikulációstempó és beszédtempó mérésére is. Ennek a kísérletnek a segítségével igazolhatjuk vagy cáfolhatjuk a H2 hipotézisünket.

A negyedik kísérlet során a harmadik kísérlettel megegyező jellemzőhalmazzal dolgoztunk, de külön végeztük el a regressziós eljárást a férfiak és a nők esetében. Tehát egyszerre alkalmaztuk mindkét vizsgált eljárási módszert. Erre a kísérletre a továbbiakban, mint „*gender+segmented*” kísérletre hivatkozunk. Ennek lényege az volt, hogy megvizsgáltuk, hogyan módosul a *baseline* regressziós eljárás pontossága, ha egyszerre alkalmazzuk az általunk javasolt két módszert, vagyis külön modellt a nők és a férfiak esetében, illetve plusz speciális jellemzőhalmaz használata, amelyek a beszéd fonéma szintű szegmentálásnak segítségével kerültek kiszámításra a 3.1 alfejezetben tárgyaltak szerint.

4 Eredmények

Az egyes kísérletek eredményeit a 2. táblázatban foglaltuk össze. A táblázat utolsó oszlopában (Rel. Vál.) a RMSE relatív változást adtuk meg a *baseline* kísérlet RMSE eredményének értékéhez képest. Azoknál a kísérleteknél (*gender* és *gender+segmented*), ahol külön modellt használtunk a nők és a férfiak esetén az eredmény alatt külön jelölve vannak a nemenként kapott eredmények is.

A 3. ábrán megadtuk az egyes kísérletek esetén az automatikus gépi eljárás által becsült BDI-értékeket az eredeti értékhez képest. Az ábrán négy kisebb ábra látható, mindegyik bal felső sarkában jeleztük, hogy melyik kísérlethez tartozik. Az ábrákon szereplő keresztek jelzik a Magyar Depressziós Adatbázisban szereplő hangmintákat, a vízszintes tengelyen leolvasható az eredeti BDI pontszámuk, míg a függőlegesen az adott kísérlet alapján becsült BDI pontszámuk. A könnyebb áttekintés érdekében mindegyik kísérlethez tartozó ábrán behúztuk az átlót, hiszen az ettől való távolság jelzi, hogy az adott minta esetében mennyit tévedett a gépi eljárás.

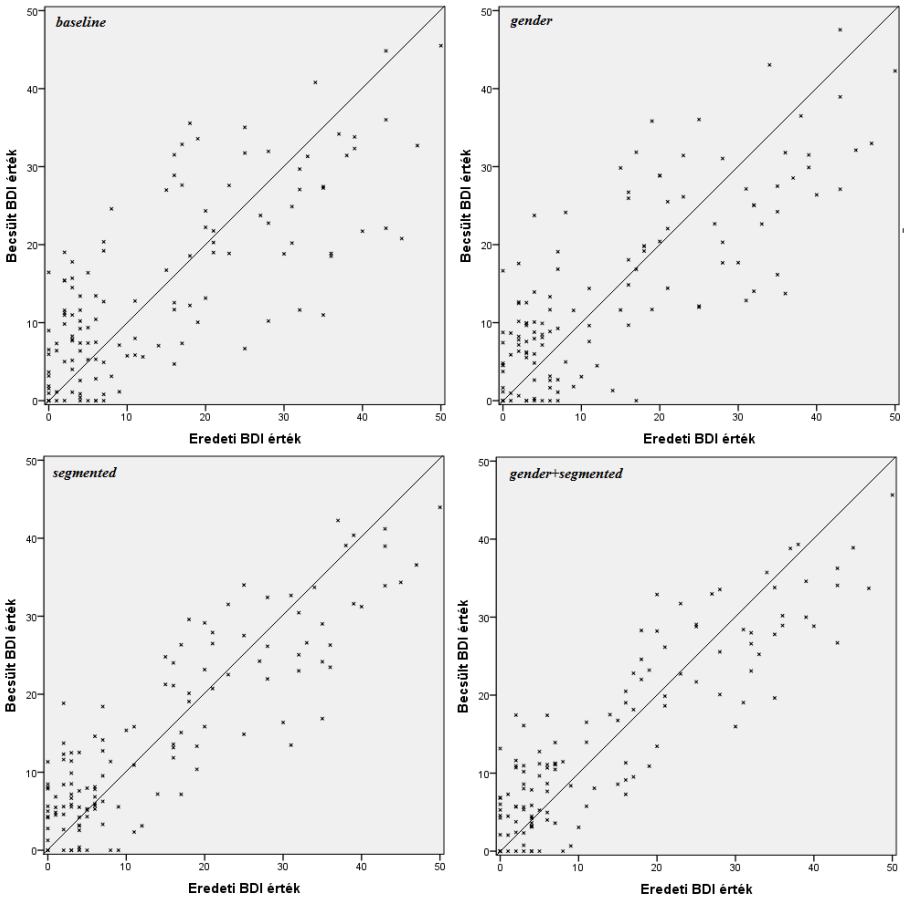
Ahhoz hogy a Magyar Depressziós Adatbázison elért regressziós eredményeink pontosságát össze lehessen hasonlítani más kutatók eredményével, beleraktuk a táblázatba az AVEC-2013 verseny győztese által használt regressziós eljárás leíró jellemzőit is (táblázatban szürke háttérrel van jelölve) [17]. Az AVEC-2013 versenyen regressziós eljárást kellett kidolgozni a megadott német nyelvű depressziós beszédadatbázis hang-

felvételeit felhasználva, az azokat bemondók BDI-II skála szerinti pontszámuk becslésére. A német adatbázisban szereplő bemondók életkora (átlag 31,5 év és 12,3 szórás) és BDI-II szerinti BDI pontszámuk eloszlása (átlag 14,9 és 11,7 szórás) nagyon hasonló a Magyar Depressziós Adatbázishoz. Természetesen ettől még ez az összehasonlítás nem tökéletes, hiszen eltérő a két adatbázis nyelve, illetve a benne lévő személyek is, mégis úgy gondoljuk, hogy jó viszonyítási alapot ad.

2. Táblázat: Az automatikus becslő rendszer pontosságának leíró jellemzői az elvégzett kísérletek esetében, kiegészítve az AVEC 2013 verseny győztesének az eredményeivel.

	MAE	**RMSE**	**Pearson corr.**	**Rel. Vál.**
AVEC 2013 győztese	*6,53*	*8,5*	*0,7*	
Baseline	7,12	9,02	0,75	
Gender	6,76 Nők: 7,07 Férfiak: 6,26	8,37 Nők: 8,7 Férfiak: 7,81	0,79 Nők: 0,68 Férfiak: 0,79	-7%
Segmented	5,31	6,6	0,87	-27%
Gender+Segmented	5,1 Nők: 5,71 Férfiak: 4,1	6,28 Nők: 6,95 Férfiak: 4,99	0,89 Nők: 0,87 Férfiak: 0,92	-30%

Az eredmények alapján kijelenthetjük, hogy mind a H1, mind a H2 hipotézisünk beigazolódott, vagyis pontosabb becslést lehet adni, ha külön női és férfi modellt használunk, illetve beszédszegmentáló használatával ezáltal pontosabb és jobb – előállított jellemzőkkel. A javulás mértéke a beszédszegmentáló használata esetében volt jelentősebb, 27%-os relatív csökkenése az átlagos négyzetes hiba gyökének. Azonban a női-férfi elkülönítéssel sem elhanyagolható a javulás mértéke, az átlagos négyzetes hiba gyökének 7%-os relatív csökkenése.



3. ábra: Az automatikus becslések értékei a négy elvégzett kísérlet esetében, összehasonlítva a minták eredeti BDI-értékeivel.

4 Konklúzió

A cikkben bemutattunk egy automatikus rendszert, ami SVR regressziós eljárással képes a beszédjel alapján megállapítani a beszélő depressziós állapotának a súlyosságát. Több kísérletet is elvégeztünk, hogy megvizsgáljuk, hogyan változik a becslés pontossága, ha eltérő módon hozzuk létre a rendszert.

A kísérletek során a Magyar Depressziós Adatbázist használtuk. Megadtunk egy *baseline* eredményt, aminek a kialakításában a korábbi, ebben a témában nyert tapasztalatainkra támaszkodtunk összhangban a nemzetközi irodalomban található eredményekkel. A rendszer pontosságának a leírására RMSE, MAE és Pearson-féle korrelációs értékeket használtunk. Így a *baseline* eredményeknek, ahol együtt kezeltük a női és a férfi mintákat, illetve a jellemzővektorok kialakításához nem használtuk fel a beszéd fonéma szintű szegmentálását, a következő értékeket kaptuk *RMSE: 9,02, MAE:*

7,12, Pearson corr.: 0,75). Ezt az eredményt összehasonlítva az AVEC 2013 győztesének az eredményével (*RMSE: 8,5, MAE: 6,53, Pearson corr.: 0,7*) megállapíthatjuk, hogy a rendszerünk bár rosszabbul teljesített (RMSE-érték alapján), de a hiba növekményének a mértéke nem számottevő, természetesen ez az összehasonlítás nem tökéletes, mivel a két kísérlet eltérő adatbázist használ. Érdemes még összevetni a *baseline* RMSE-értéket az adatbázisban található személyek BDI-értékeinek a szórásával, ami 13,5, mivel ezt az értéket kapnánk, ha minden személyhez az átlagos BDI-értéket rendelnénk hozzá, vagyis a rendszer RMSE-ben kifejezve 4,48 hibapont értékkel teljesít jobban, mint az elvárt minimum.

A munka elején két hipotézist fogalmaztunk meg, miszerint a rendszer pontossága javul, ha külön kezeljük a női és férfi mintákat (H1), illetve ha automatikus beszédszegmentálót használva, a bemondások fonéma szintű szegmentálásnak segítségével, egy jobb, pontosabb, kibővített jellemzőhalmazt alkalmazunk (H2).

Mindkét hipotézisünket igazoltuk, ugyanis 7%-os relatív javulást értünk el az RMSE-értékben, ha külön kezeltük a férfi és női mintákat, illetve 27%-os relatív javulást értünk el, ha felhasználtuk a jellemzők előállításánál a beszéd fonéma szintű szegmentálását a *baseline* eredményekhez képest. Az eredmények alapján a szegmentálás alapú jellemző számítás tűnik fontosabbnak a pontosság szempontjából, ugyanakkor tény, hogy a nemek szétválasztása esetén csökkent a tanító minták száma az így kialakított két külön rendszerben, így ennek a tükrében a 7%-os relatív javulás mindenképpen jelentős.

Legvégül megvizsgáltuk, hogyan változik a pontosság, ha a két vizsgált módszert ötvözzük *(RMSE: 6,28, MAE: 5,1, Pearson corr.: 0,89)*. RMSE-ben mérve a *baseline* eredményhez képest 30%-os relatív javulást értünk el, míg maga a hibaérték abszolút értelemben véve is kifejezetten alacsonynak mondható, így akár ez a módszer alkalmas lehet egy depressziós állapotot diagnosztizáló rendszer alapjának.

Továbbiakban tervezzük más kevésbé elterjedt jellemzők felhasználásával is megvizsgálni a rendszerünk pontosságának változását. Tervezzük, hogy módszerünket kipróbáljuk más adatbázisokon is. Illetve az adatbázisunkat folyamatosan bővítjük.

Köszönetnyilvánítás

A kutatást támogatta az ESA ügynökség COALA projekt: Psychological Status Monitoring by Computerised Analysis of Language phenomena (COALA) (AO-11-Concordia).

Bibliográfia

1. Beck, A.T., Steer, R.A., Ball, R., Ranieri, W.F., (1996). Comparison of beck depression inventories-ia and-ii in psychiatric outpatients. *J. Pers. Assess.* 67, 588–597.
2. Boersma, P., (2002). Praat, a system for doing phonetics by computer. *Glot international*, 5(9/10), pp.341-345.

3. Brendel, R.W., Wei, M., Lagomasino, I.T., Perlis, R.H., Stern, T.A., (2010). Care of the suicidal patient. *Massachusetts General Hospital Handbook of General Hospital Psychiatry*, 6th ed. W.B. Saunders, Saint Louis, pp. 541–554.
4. Chang, C. C., & Lin, C. J. (2011). LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3), 27.
5. Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273-297.
6. Cummins, N., Scherer, S., Krajewski, J., Schnieder, S., Epps, J., & Quatieri, T. F. (2015). A review of depression and suicide risk assessment using speech analysis. *Speech Communication*, 71, 10-49.
7. Hamilton, H., (1960). HAMD: a rating scale for depression. *Neurosurg. Psych*. 23, 56–62.
8. Kiss, G., Sztahó, D. and Vicsi, K., (2013), December. Language independent automatic speech segmentation into phoneme-like units on the base of acoustic distinctive features. In *Cognitive Infocommunications (CogInfoCom)*, 2013 IEEE 4th International Conference on (pp. 579-582). IEEE.
9. Kiss, G. and Vicsi, K., (2014). Physiological and cognitive status monitoring on the base of acoustic-phonetic speech parameters. In *International Conference on Statistical Language and Speech Processing* (pp. 120-131). Springer International Publishing.
10. Kraepelin, E., (1921). Manic depressive insanity and paranoia. *J. Nerv. Ment. Dis*. 53, 350.
11. Marcus, M., Yasamy, M. T., van Ommeren, M., Chisholm, D. and Saxena, S. (2012). Depression: A global public health concern. *WHO Department of Mental Health and Substance Abuse*, 1, 6-8.
12. Mathers, C. D., & Loncar, D. (2006). Projections of global mortality and burden of disease from 2002 to 2030. *Plos med*, 3(11), e442.
13. Mundt, J. C., Snyder, P. J., Cannizzaro, M. S., Chappie, K., & Geralts, D. S. (2007). Voice acoustic measures of depression severity and treatment response collected via interactive voice response (IVR) technology. *Journal of neurolinguistics*, 20(1), 50-64.
14. Mundt, J. C., Vogel, A. P., Feltner, D. E., & Lenderking, W. R. (2012). Vocal acoustic biomarkers of depression severity and treatment response. *Biological psychiatry*, 72(7), 580-587.
15. Nilsonne, A., (1988). Speech characteristics as indicators of depressive illness. *Acta Psych. Scand*. 77, 253–263.
16. Smola, A., & Vapnik, V. (1997). Support vector regression machines. *Advances in neural information processing systems*, 9, 155-161.
17. Williamson, J. R., Quatieri, T. F., Helfer, B. S., Horwitz, R., Yu, B., & Mehta, D. D. (2013). Vocal biomarkers of depression based on motor incoordination. In *Proceedings of the 3rd ACM international workshop on Audio/visual emotion challenge* (pp. 41-48). ACM.

Neurális hálók tanítása valószínűségi mintavételezéssel nevetések felismerésére

Gosztolya Gábor[1,2], Grósz Tamás[2], Tóth László[1],
Beke András[3], Neuberger Tilda[3]

[1] MTA-SZTE Mesterséges Intelligencia Kutatócsoport
Szeged, Tisza Lajos krt. 103., e-mail: ggabor@inf.u-szeged.hu
[2] Szegedi Tudományegyetem, Informatikai Intézet
Szeged, Árpád tér 1.
[3] MTA Nyelvtudományi Intézet
Budapest, Benczúr u. 33., e-mail: beke.andras@nytud.mta.hu

Kivonat Mikor a feladat spontán beszédben nevetések előfordulásait megtalálni, kézenfekvő megközelítés a beszédfelismerés feladatkörében gyakran használt technikákat alkalmazni. Például becsülhetjük a nevetés valószínűségét lokálisan, a keretek szintjén, mely valószínűségbecsléseket szolgáltathatja például egy mély neurális háló. Ugyanakkor a hangfelvételeknek csak kis része (néhány százaléka) felel meg nevetésnek; a többit beszéd, csend, háttérzajok, stb. teszik ki. Ez azt eredményezi, hogy a mély neurális hálót olyan adatokon tanítjuk, melyeknél az osztályelőfordulás szélsőségesen kiegyensúlyozatlan. Jelen cikkünkben a valószínűségi mintavételezés (*probabilistic sampling*) nevű eljárást alkalmaztuk a mély neurális hálók tanítása során, mellyel 7%-os relatív hibacsökkentést tudtunk elérni a keretszintű F_1 pontosságértékeket tekintve.

Kulcsszavak: nevetésdetektálás, mély neurális hálók, tanítópélda-mintavételezés

1. Bevezetés

Az emberiséget mindig is érdekelte viselkedésének alapvető megértése, előrejelezhetőségének lehetősége. Az elmúlt évtizedekben köszönhetően a technikai fejlődésnek (főként az agyi képalkotó eljárásoknak, a hang- és videórögzítésnek, valamint ezek gyors feldolgozhatóságának) egyre mélyebb ismereteink vannak az emberi viselkedésről. A beszédtudomány fókuszában főként annak vizsgálata áll, hogy hogyan viselkedünk a társas kommunikáció során. Ezen viselkedés feltérképezésnek az egyik kulcseleme a non-verbális kommunikáció vizsgálata a társalgás során. Egyes feltételezések szerint a non-verbális kommunikáció közel kétharmadát teszi ki a teljes kommunikációnak [1], és használata kevésbé kontrollált, így vizsgálatával alapvető viselkedési mintázatokat lehet kimutatni. A

Grósz Tamást az Emberi Erőforrások Minisztériuma ÚNKP-16-3 kódszámú Új Nemzeti Kiválóság Programja támogatta.

non-verbális kommunikáció során folyamatos nem-lexikális elemek küldése és fogadása történik az egyes emberek között. Modalitásukat tekintve ezek különfélék lehetnek, mint a testtartás, a szemmozgás vagy a non-verbális vokális elemek. Elsőleges szerepük a magatartás és az érzelmek kifejezésében van [2]. Mindemellett fontos szerepet töltenek be a dialógusok szerveződésében [3], illetve sok szempontból tükrözik személyiségünket [4].

A non-verbális jelek további két csoportra oszthatók: vizuális és vokális [5,6]. A vokális non-verbális jelek közé tartoznak a paralingvisztikai jelek (pl. zöngeminőség, hangerő), illetve a non-verbális vokalizációk (pl. nevetés, sóhajtás, kitöltött szünetek) [7,8]. Jelen munka a nevetések automatikus felismerésére koncentrál, mivel maga a nevetés, mint non-verbális vokalizációs elem, az egyik kulcseleme a társalgás során mutatott viselkedés feltérképezésének, illetve modellezésének.

Korábbi munkáinkban [9,10] beszédszegmensek osztályozásával (nevetés vagy szöveg/csend) foglalkoztunk, és az irodalomban is számos ilyen munkával találkozhatunk (pl. [11,12]). Egy másik elterjedt megközelítésben mind a modelltanítás, mind a kiértékelés kizárólag a keretek szintjén történik (pl. [13,14,15]). A valós alkalmazásokhoz azonban közelebb áll az a megközelítés, melyben spontán beszédben akarjuk meghatározni azokat a szegmenseket, melyek nevetést tartalmaznak. Kézenfekvő, ha ekkor a beszédfelismerés területéről veszünk át eszközöket, például a keretszintű valószínűségbecsléseket egy rejtett Markov modell (Hidden Markov model, HMM) segítségével kombináljuk. Magukat a valószínűségbecsléseket előállíthatjuk Gauss keverékmodellekkel (Gaussian Mixture Models, GMM), de neurális hálókkal (Artificial Neural Networks, ANN) vagy mély neurális hálókkal (Deep Neural Networks, DNN) is.

Akusztikus modellünket tehát keretszintű jellemzővektorokon tanítjuk, melyek a két kézenfekvő osztály (nevetés illetve nem-nevetés, beleértve a beszédet, csendet, torokköszörülést, különböző háttérzajokat stb.) valamelyikébe tartoznak. Egy lényeges különbség azonban a beszédfelismerés feladatához képest, hogy a két osztályhoz tartozó példák száma nagyon kiegyensúlyozatlan: tipikusan a keretek 4-6%-a tartalmaz nevetést [7,8,16]. Ez egy diszkriminatív osztályozó (például egy mély neurális háló) tanítása során azt jelenti, hogy az az egyik osztályból lényegesen több példát lát, így azt jobban képes megtanulni, míg a másik osztály súlyosan alulreprezentált. Ez kezelhető a gyakoribb osztályokba tartozó példák egy részének elhagyásával, azonban ez nyilvánvalóan csökkenti az adott osztály variabilitását. A másik megközelítés, hogy (amennyiben nem tudunk további, a ritkább osztályokba tartozó példákat szerezni vagy generálni) egyes tanítópéldákat gyakrabban használunk a tanítás során.

Jelen cikkünkben egy, az utóbbi kategóriába tartozó tanítási eljárást alkalmazunk nevetésfelismerésre tanított mély neurális hálók esetében. Először bemutatjuk az alkalmazott módszert (*valószínűségi mintavételezés*, [17]), majd elemezzük, hogy alkalmazása hogyan befolyásolja a neurális háló által generált valószínűségbecslések rejtett Markov modellben való alkalmazását. Ezután leírjuk a kísérleti környezetet (a felhasznált adatbázist, a pontosságmetrikákat és a DNN paramétereit), végül bemutatjuk és elemezzük az eredményeket.

2. Valószínűségi mintavételezés

Mint az osztályozó módszerek általában, a neurális hálók is érzékenyek arra, ha az egyes osztályokhoz nem egyenletesen állnak rendelkezésre tanítópéldák. Ilyen esetekben hajlamosak pontatlan valószínűségbecsléseket adni az alulreprezentált osztályokhoz tartozó példákra. Ennek kezelésére talán a legegyszerűbb megközelítés, ha a gyakoribb osztályokhoz tartozó tanítópéldák számát redukáljuk; ekkor azonban nyilvánvalóan információt is veszítünk, mely az osztályozási pontosság csökkenéséhez is vezethet. Egy másik megközelítés, ha inkább gyakrabban használjuk a ritkábban előforduló osztályok tanítópéldáit. Egy matematikailag jól meghatározott ilyen tanítási stratégia a valószínűségi mintavételezés (*probabilistic sampling*, [17,18]). Ennek során a következő tanítópéldát egy kétlépéses eljárásban választjuk ki: először a példa *osztályát* határozzuk meg valamely valószínűségi eloszlást követve, majd választunk egy tanítópéldát az adott osztályból. Az osztályok kiválasztásának valószínűségére az alábbi képlet szolgál:

$$P(c_k) = \lambda \frac{1}{K} + (1 - \lambda) Prior(c_k), \tag{1}$$

ahol $Prior(c_k)$ a k. osztály (c_k) előzetes (prior) valószínűsége, K az osztályok száma, míg $0 \leq \lambda \leq 1$ egy paraméter. $\lambda = 0$ esetén ez a képlet az eredeti osztályeloszlást adja, míg $\lambda = 1$ az egyenletes eloszláshoz vezet, melyet követve a tanítás során minden osztályból közelítőleg ugyanannyi példát használunk fel. Köztes λ értékeket használva lineárisan képezünk átmenetet a két eloszlás között.

Beszédfelismerés során ritkán használnak tanítópélda-mintavételezést, melynek véleményünk szerint több oka is van. Egyrészt a tanító adatbázisok gépi tanulási szempontból igen nagynak számítanak, így egy DNN kellően pontos modellt képes építeni az egyes fonémaállapotokról (melyek az osztályoknak felelnek meg). Egy további ok szerintünk, hogy az egyes osztályokhoz tartozó példák eloszlása relatíve egyenletes. (Ezt tovább erősíti a kontextusfüggő állapotmodellezés [19,20,21] alkalmazása, melynek egyik célja épp annak garantálása, hogy minden osztályhoz kellő számú tanítópélda álljon rendelkezésre.) Érdemes kitérnünk García-Moral és tsai [22] igen részletes tanulmányára, melyben tanítópéldákat hagytak el a gyakoribb osztályokból. Habár ezzel lényegesen fel tudták gyorsítani a neurális háló tanítását, beszédfelismerő rendszerük pontossága valamelyest csökkent. Tóth és Kocsor 2005-ben alkalmazták a fönt ismertetett valószínűségi mintavételezési módszert egy kisszótáras, izolált szavas felismerő akusztikus modelljének (sekély neurális háló) tanítására. García-Moral cikkével ellentétben ők ezzel növelni is tudták a felismerés pontosságát.

Ezek a tanulmányok beszédfelismerési kontextusban mintavételezték a tanítópéldákat az akusztikus modell tanítása során, mely feladatban az osztályok eloszlásának különbsége minimális. Ugyanakkor nevetés és a hasonló nemverbális hangjelenségek (pl. kitöltött szünetek) felismerése esetén az osztályok megoszlása sokkal kiegyensúlyozatlanabb, hiszen a felvételeknek csak egy töredéke (nevetések esetén pl. tipikusan 4-6%-a) felel meg a keresett jelenségnek. Ebben az esetben joggal várhatjuk, hogy valamely mintavételezési eljárás alkalmazása a tanítás során jelentősen javítja a detektálás hatékonyságát.

1. táblázat. A BEA adatbázis felhasznált részének néhány jellemzője

		Halmaz			Összes
		Tanító	Fejlesztési	Teszt	felvétel
Felvételek összhossza (p:mp)		100:07	20:32	26:57	147:36
Nevetések	összhossza (p:mp)	7:53	1:55	2:14	12:01
	aránya	7,8%	9,3%	8,3%	8,1%
	gyakorisága (1/p)	5,21	5,07	5,53	5,25
	átlagos hossza (ms)	903	1106	901	930

2.1. Valószínűségi mintavételezés alkalmazása rejtett Markov modellben

Egy szokásos rejtett Markov modell minden keretszintű x_t megfigyelésvektorhoz és minden c_k állapothoz $p(x_t|c_k)$ valószínűség-becsléseket vár bemenetként. Mivel a neurális hálók a $P(c_k|x_t)$ értékeket becslik, a várt $p(x_t|c_k)$ értékeket a Bayes-tétel alkalmazásával kaphatjuk meg. Így egy HMM/ANN vagy HMM/DNN hibrid modell használatakor a neurális háló keretszintű kimeneteit el kell osztanunk a megfelelő osztály a priori valószínűségével ($P(c_k)$). Ezzel a kívánt $p(x_t|c_k)$ becsléseket kapjuk egy konstans szorzótól eltekintve, amely konstans szorzót azonban (a Viterbi keresés során alkalmazott maximalizálás miatt) figyelmen kívül hagyhatjuk.

Ugyanakkor Tóth és Kocsor [18] megmutatták, hogy amennyiben neurális hálóinkat $\lambda = 1$ paraméterrel tanítjuk (azaz egyenletes osztályeloszlást használunk), azok a $p(x_t|c_k)$ értékeket fogják becsülni (ismét egy konstansszorzótól eltekintve, amelyet megint figyelmen kívül hagyhatunk). Eszerint tehát $\lambda = 1$ paraméterérték használata esetén a háló által szolgáltatott valószínűségbecsléseket már nem kell tovább transzformálnunk, hanem azokat közvetlenül használhatjuk egy rejtett Markov modellben.

Elviekben tehát vagy $\lambda = 0$ paraméterezést kellene használnunk, és osztanunk az osztályok prior valószínűségeivel ($P(c_k)$), vagy $\lambda = 1$-et, és nem alkalmazni a Bayes-formulát. A gyakorlatban azonban a valószínűségbecslések nem pontosak, így jobb eredményeket kaphatunk köztes λ paraméterértékek használatával. Tóth és Kocsor cikkében [18] szintén köztes λ értékek adódtak optimálisnak. Mivel ebben az esetben nem egyértelmű, hogy érdemes-e alkalmaznunk a Bayes-formulát, mi mind a két stratégiát ki fogjuk próbálni.

3. Kísérletek

3.1. Adatbázis

Kísérleteinket a BEA adatbázis [23] egy részhalmazán végeztük. A BEA a legnagyobb magyar szabadon elérhető beszédadatbázis, melynek teljes felvételhossza 260 óra összesen 280 beszélőtől, hangszigetelt stúdiókörülmények között rögzítve.

Az adatbázis egy lényeges tulajdonsága, hogy spontán beszédet tartalmaz, mely fontos kritériuma annak, hogy nevetést tartalmazzon. Kísérleteinket 62 felvételen végeztük; 42-n tanítottuk az akusztikus neurális hálókat, 10-et fejlesztési halmazként, 10-et pedig tesztként használtunk. A tanító rész összesen 100, a fejlesztési halmaz 21, míg a teszthalmaz összesen 27 perc hosszú volt.

Az 1. táblázat tartalmazza a kísérletekhez használt felvételek néhány nevetés-specifikus jellemzőjét. Látható, hogy habár a fejlesztési és a teszthalmazt véletlenszerűen választottuk és csupán tíz-tíz felvételből állnak, elég jól reprezentálják a teljes hanganyagot. Ezen az adathalmazon a nevetésnek annotált részek aránya az irodalomban jellemzően említett 4-6%-nál valamivel nagyobbnak, 8% körülinek adódott, mely azonban még így is csak a felvételek töredéke. Az átlagos nevetéshossz majdnem egy másodperc, amely meglepően magas, azonban más cikkekben (pl. [16]) is hasonló értékekkel találkozhatunk.

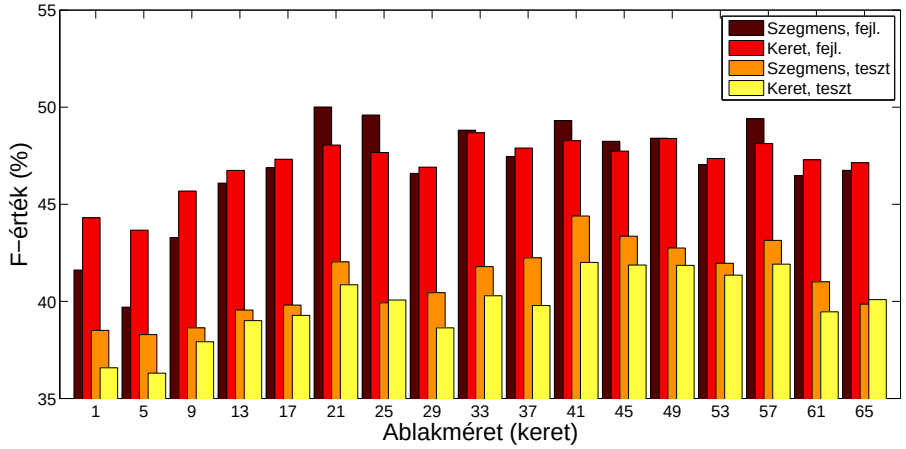
3.2. Kiértékelés

A nevetésdetektálás feladatánál nincs olyan egyértelműen elterjedt kiértékelési metrika, mint amilyen a szószintű hiba a beszédfelismerés területén. A legegyszerűbb megoldás, ha keretszintű pontosságot vizsgálunk; az osztályozási pontosság azonban közismerten nem rangsorolja jól a modelleket, ha az osztályeloszlás nagyon kiegyensúlyozatlan. Ennek egy finomításaként értékelhető az a gyakran használt megközelítés (ld. pl. [13,14,15]), melyben a keresett jelenséghez tartozó, keretszintű osztályvalószínűségekre meghatározzuk a ROC görbét, valamint a görbe alatti területet (Area Under Curve, AUC). Ennél életszerűbb kritériumnak gondoljuk ugyanakkor, hogy a valószínűségbecslésekből egy rejtett Markov modell segítségével szegmensszintű (kezdet- és végponttal rendelkező) előfordulás-hipotéziseket alkossunk, és a modellt ezek alapján értékeljük.

Tekintve, hogy a nevetésfelismerés egy standard információ-visszakeresési (Information Retrieval, IR) feladat, szokásos IR metrikákat számoltunk a modellek pontosságának mérésére: pontosságot (*precision*), fedést (*recall*) és F-értéket (*F-measure* vagy F_1). Ezeket csak a nevetés osztályra számítottuk ki, azonban két megközelítést is alkalmaztunk. Az egyikben nevetés*szegmenseket* vizsgáltunk (egy annotált szegmenst akkor tekintettünk megtaláltnak, ha egy hipotézis szegmens metszette a referencia annotációt és a két szegmens közepe maximum 0,5 másodpercre esett egymástól [24]). A másikban a rejtett Markov modell kimenetét keretszintre konvertáltuk, és a három metrikát a keretekre számítottuk ki [16].

3.3. A neurális háló és paraméterei

Saját neurálisháló-implementációnkat használtuk, mellyel korábban sok különböző feladaton értünk el jó eredményeket (pl. [25,26,27,28]). A neurális hálókat keretszinten tanítottuk, az FBANK jellemzőkészletet használva, amely 40 Mel szűrősor energiáiból, illetve azok első- és másodrendű deriváltjaiból áll [29]. Alkalmaztuk azt a fonémaosztályozás esetén bevett megoldást is, hogy a szomszédos



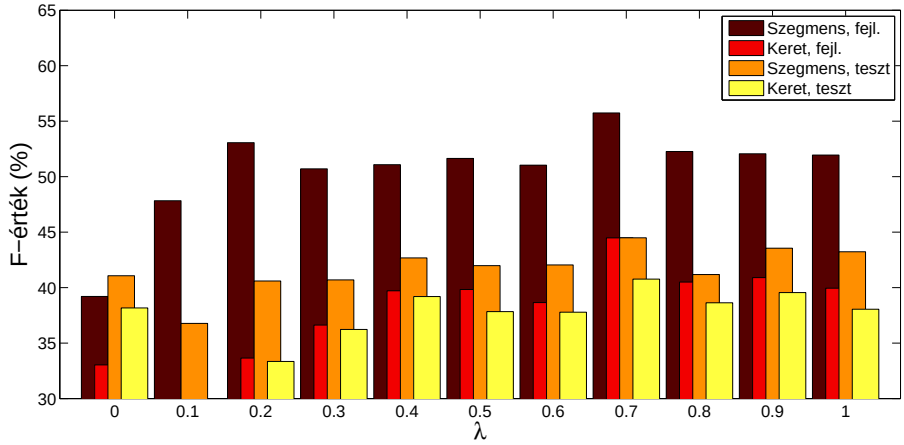
1. ábra. Átlagos F_1-értékek a tanításra használt mozgó ablak méretének függvényében

keretek jellemzővektorait is felhasználtuk az egyes keretek osztályozása során. Az alkalmazott neurális hálók előzetes tesztek eredményei alapján öt rejtett réteggel rendelkeztek, melyek mindegyikében 256 rectifier függvényt alkalmazó neuron volt, míg a kimeneti rétegben softmax függvényt használtunk. A súlyokat L2 regularizációval tartottuk kordában.

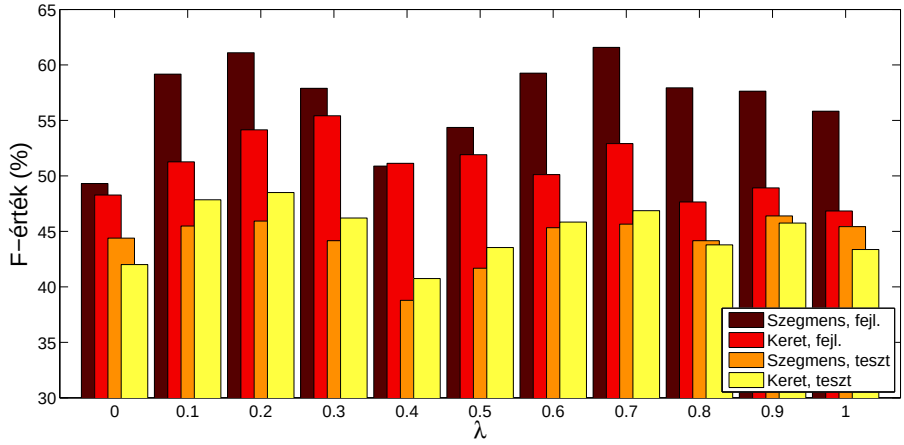
Mivel a neurális háló tanítása sztochasztikus folyamat (köszönhetően a súlyok véletlen inicializálásának), minden tesztelt λ paraméterváltozatra öt-öt hálót tanítottunk, és a kapott pontosságértékeket kiátlagoltuk. Salamin és tsai. [16] dolgozatát követve keretszintű nyelvi modellt számítottunk a tanítási halmazon; ennek súlyát minden neurális hálóra külön-külön, a fejlesztési halmazon határoztuk meg. Külön nyelvimodell-súlyt állapítottunk meg annak függvényében is, hogy a pontosságot szegmens- vagy keretszinten mértük-e.

Viszonyítási alapként teljes mintavételezéssel tanított mély neurális hálók szolgáltak. A tanítást a kereteken vett csúszó ablakokon végeztük, melyek optimális méretét előzetes tesztekkel határoztuk meg. Ennek során a csúszó ablak $1, 5, \ldots, 65$ keret széles volt, a DNN által szolgáltatott keretszintű valószínűségbecsléseket pedig a Bayes-formulával korrigáltuk. Az eredmények az 1. ábrán láthatóak; a fejlesztési halmazon mért szegmens- és keretalapú F_1-értékek alapján az eredmények alapján a mozgó ablak méretét a továbbiakban 41 keretnek választottuk.

A valószínűségi mintavételezés λ paraméterét a $0 < \lambda \leq 1$ intervallumban teszteltük, $0,1$-es lépésközt használva, minden λ értékre öt hálót tanítva. Az optimális λ értéket a fejlesztési halmazon határoztuk meg. Teszteltük, hogy a posterior valószínűségeket érdemes-e a Bayes-tétel alkalmazásával transzformálnunk, vagy inkább az eredeti értékeket érdemes használnunk. Fontos észrevétel, hogy ehhez nem volt szükséges új hálókat tanítanunk.



2. ábra. Átlagos F_1-értékek a valószínűségi mintavételezés λ paraméterének függvényében, a Bayes-tétel alkalmazása nélkül



3. ábra. Átlagos F_1-értékek a valószínűségi mintavételezés λ paraméterének függvényében, a Bayes-tétel alkalmazása után

3.4. Eredmények

A 2. és 3. ábrán láthatóak az átlagos F_1 értékek a λ paraméter függvényében. Ahogyan az várható volt, az eredeti posterior értékek használata esetén (ld. 2. ábra) a magasabb λ értékek ($\lambda \geq 0,7$), míg a Bayes-formulával korrigált valószínűségbecslések esetén (ld. 3. ábra) inkább az alacsonyabb λ értékek mellett mért pontosságok adódtak valamivel magasabbnak. Látható, hogy mindkét esetben köztes ($0 < \lambda < 1$) λ értékek adódtak optimálisnak. Ugyanakkor az eredeti posteriorok használatával nem sikerült elérni a referencia-értékeket (amelyek a Bayes-képlet alkalmazásával, viszont teljes mintavételezés mellett születtek).

2. táblázat. A valószínűségi mintavételezési eljárással kapott optimális átlagos F_1-értékek

Kiértékelés szintje	Halmaz	Priorokkal osztás	Opt. λ	Pontosság Prec.	Rec.	F_1	Relatív hibacsökk.
Szegmens	Fejlesztési	nem	0,7	53,51%	58,46%	55,74%	12,68%
		igen	0,7	59,36%	64,03%	61,58%	24,21%
		igen	—	41,11%	62,50%	49,31%	—
	Teszt	nem	0,7	43,85%	45,37%	44,49%	0,16%
		igen	0,7	45,96%	45,37%	45,65%	2,25%
		igen	—	39,42%	51,55%	44,40%	—
Keret	Fejlesztési	nem	0,7	61,45%	34,96%	44,48%	-7,33%
		igen	0,3	46,49%	68,60%	55,42%	13,82%
		igen	—	38,02%	66,53%	48,27%	—
	Teszt	nem	0,7	51,60%	33,81%	40,77%	-2,14%
		igen	0,3	36,09%	64,22%	46,20%	7,23%
		igen	—	30,94%	66,14%	42,01%	—

Az 2. táblázat foglalja össze a legjobb pontosságértékeket a fejlesztési-, és az azonos meta-paraméterekkel született pontosságértékeket a teszthalmazon. A táblázatban az átlagos F_1 érték mellett a pontosságot és a fedést is feltüntettük. Látható, hogy az F_1 értékeken szegmensszinten lényegesen sikerült javítani a fejlesztési halmazon, azonban a teszthalmazra ennek csak egy töredékét sikerült átvinni. A keretek szintjén enyhe csökkenést tapasztalhatunk, mikor a mély neurális háló valószínűségbecsléseit közvetlenül alkalmaztuk a rejtett Markov modellben; a Bayes-tétel alkalmazását követően azonban az F_1-értékek a teszthalmazon is jelentősen javultak: a teszthalmazon 42%-os viszonyítási értékről 46% fölé nőttek, mely 7%-os relatív hibacsökkentést jelent.

A referencia esetekben a fedés jóval magasabb volt, mint a pontosság, ami sok fals pozitív találatra utal. Valószínűségi mintavételezést használva szegmensszinten a két érték szinte tökéletesen kiegyensúlyozott, keretszinten azonban eltérések tapasztalhatóak. Ez arra utal, hogy a rejtett Markov modell ugyan elég jó pontossággal megtalálja a nevetés-előfordulásokat, a szegmensek határait illetően azonban bizonytalan.

A mély hálók kimeneteit változatlan formában használva keretszinten magas pontosságot és alacsony fedést, míg a Bayes-tétel után relatíve alacsony pontosságot és magas fedést láthatunk. Ez elég logikus: a mély háló vélhetően alapvetően alacsony valószínűségértékeket becsült a nevetés osztályra, melyeket közvetlenül használva a rejtett Markov modellben csak az egyértelműen nevetést tartalmazó keretek lettek azonosítva. Az osztályok a priori valószínűségeivel osztva a hálók kimenetét azonban változik a helyzet: mivel a nevetés osztálynak alacsony az a priori valószínűsége, a beszédet és csendet jelentő osztálynak pedig elég magas, a Bayes-tétel alkalmazásával a nevetésre adott becsléseinket

nagymértékben megnöveljük, míg a másik osztályét csak alig. Így vélhetően a nevetést tartalmazó szegmensek környezetében található kereteket is nevetésnek azonosítjuk, mely a szegmensszintű pontosságértékeket nem változtatja meg, keretszinten azonban csökkenti a fals negatív és növeli a fals pozitív találatok arányát.

3.5. Konklúzió

Jelen dolgozatban spontán beszédben kerestük nevetések előfordulását egy rejtett Markov modell/mély neurális háló keretrendszerben. Mivel a nevetés a hanganyagnak csak mintegy 8%-át tette ki, a tanítópéldák osztályeloszlása egyenetlen volt, így mély neurális hálónk tanítása szuboptimális volt. Kísérletileg megmutattuk, hogy a tanítás javítható a tanítópéldák újra-mintavételezésével. A valószínűségi mintavételezés nevű eljárás használatával a keretszintű hibát 7%-kal tudtuk csökkenteni egy magyar nyelvű, spontán beszédet tartalmazó adatbázison.

Hivatkozások

1. Hogan, K.: Can't Get Through: Eight Barriers to Communication. Pelican Publishing (2003)
2. Halberstadt, A.G.: Family socialization of emotional expression and nonverbal communication styles and skills. Journal of personality and social psychology **51**(4) (1986) 827
3. Johannesen, R.L.: The emerging concept of communication as dialogue. (1971)
4. Isbister, K., Nass, C.: Consistency of personality in interactive characters: verbal cues, non-verbal cues, and user characteristics. International journal of human-computer studies **53**(2) (2000) 251–267
5. Glenn, P.: Laughter in interaction. Cambridge University Press, Cambridge, UK (2003)
6. Hámori, A.: Nevetés a társalgásban. In Laczkó, K., Tátrai, S., eds.: Elmélet és módszer. ELTE Eötvös József Collegium, Budapest, Hungary (2014) 105–129
7. Holmes, J., Marra, M.: Having a laugh at work: How humour contributes to workplace culture. Journal of Pragmatics **34**(12) (2002) 1683–1710
8. Neuberger, T.: Nonverbális hangjelenségek a spontán beszédben. In Gósy, M., ed.: Beszéd, adatbázis, kutatások. Akadémiai Kiadó, Budapest (2012) 215–235
9. Gosztolya, G., Beke, A., Neuberger, T.: Nevetések automatikus felismerése mély neurális hálók használatával. In: MSZNY, Szeged (2016) 122–133
10. Gosztolya, G., Beke, A., Tóth, L., Neuberger, T.: Laughter classification using deep rectifier neural networks with a minimal feature subset. Archives of Acoustics **41**(4) (2016) 669–682
11. Knox, M.T., Mirghafori, N.: Automatic laughter detection using neural networks. In: Proceedings of Interspeech, Antwerp, Belgium (2007) 2973–2976
12. Neuberger, T., Beke, A.: Automatic laughter detection in spontaneous speech using GMM–SVM method. In: TSD. (2013) 113–120
13. Gupta, R., Audhkhasi, K., Lee, S., Narayanan, S.S.: Detecting paralinguistic events in audio stream using context in features and probabilistic decisions. Computer, Speech and Language **36**(1) (2016) 72–92

14. Kaya, H., Ercetin, A., Salah, A., Gürgen, S.: Random forests for laughter detection. In: WASSS. (2013)
15. Brueckner, R., Schuller, B.: Social signal classification using deep BLSTM recurrent neural networks. In: Proceedings of ICASSP. (2014) 4856–4860
16. Salamin, H., Polychroniou, A., Vinciarelli, A.: Automatic detection of laughter and fillers in spontaneous mobile phone conversations. In: Proceedings of SMC. (2013) 4282–4287
17. Lawrence, S., Burns, I., Back, A., Tsoi, A., Giles, C.: Chapter 14: Neural network classification and prior class probabilities. In: Neural Networks: Tricks of the Trade. Springer (1998) 299–313
18. Tóth, L., Kocsor, A.: Training HMM/ANN hybrid speech recognizers by probabilistic sampling. In: Proceedings of ICANN. (2005) 597–603
19. Young, S.J., Odell, J.J., Woodland, P.C.: Tree-based state tying for high accuracy acoustic modelling. In: HLT. (1994) 307–312
20. Wang, W., Tang, H., Livescu, K.: Triphone state-tying via deep canonical correlation analysis. In: Interspeech, San Francisco, USA (Sep 2016) 3444–3448
21. Gosztolya, G., Grósz, T., Tóth, L., Imseng, D.: Building context-dependent DNN acousitc models using Kullback-Leibler divergence-based state tying. In: ICASSP, Brisbane, Ausztrália (2015) 4570–4574
22. García-Moral, A.I., Solera-Urena, R., Peláez-Moreno, C., de María, F.D.: Data balancing for efficient training of hybrid ANN/HMM Automatic Speech Recognition systems. IEEE Trans. ASLP **19**(3) (2011) 468–481
23. Gósy, M.: Bea a multifunctional hungarian spoken language database. The Phonetician **105**(106) (2012) 50–61
24. : NIST Spoken Term Detection 2006 Evaluation Plan. http://www.nist.gov/speech/tests/std/docs/std06-evalplan-v10.pdf. (2006)
25. Tóth, L.: Phone recognition with hierarchical Convolutional Deep Maxout Networks. EURASIP Journal on Audio, Speech, and Music Processing **2015**(25) (2015) 707–710
26. Gosztolya, G.: On evaluation metrics for social signal detection. In: Interspeech, Drezda, Németország (2015) 2504–2508
27. Grósz, T., Busa-Fekete, R., Gosztolya, G., Tóth, L.: Assessing the degree of nativeness and Parkinson's condition using Gaussian Processes and Deep Rectifier Neural Networks. In: Interspeech, Drezda, Németország (2015) 1339–1343
28. Kovács, Gy., Tóth, L.: Joint optimization of spectro-temporal features and Deep Neural Nets for robust automatic speech recognition. Acta Cybernetica **22**(1) (2015) 117–134
29. Young, S., Evermann, G., Gales, M.J.F., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The HTK Book. Cambridge University Engineering Department, Cambridge, Anglia (2006)

Élő labdarúgó-közvetítések gépi feliratozása

Tarján Balázs[1], Szabó Lili[3], Balog András[2], Halmos Dávid[3],
Fegyó Tibor[1,3] és Mihajlik Péter[1,2]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
tarjanb@tmit.bme.hu
[2] THINKTech Kutatási Központ Nonprofit Kft.
mihajlik@thinktech.hu
[3] SpeechTex Kft.
tfegyo@speechtex.com

Kivonat: Fontos cél, hogy egyre több televíziós műsor legyen akadálymentesen hozzáférhető siket- és nagyothalló nézők számára is. Az élő labdarúgó-mérkőzések kiemelten nagy népszerűségnek örvendenek, így feliratozásuk sok ember életét könnyíti meg. A jelenleg alkalmazott kézi feliratozásnál azonban sok információ elveszik a kommentárból. Erre nyújthat megoldást a gépi feliratozás, mely képes lépést tartani az élő beszéd tempójával. Technikai szempontból azonban problémát jelent az idegennyelvű személynevek felismerése, melyek gyakoriak a labdarúgó-közvetítésekben, így nagyban befolyásolják a feliratozás minőségét. Cikkünkben különböző szótárbővítési eljárások hatékonyságát vetjük össze. Ezen módszerek előnye, hogy segítségükkel egy már betanított felismerő modell szótárát a kívánt feladathoz lehet adaptálni. Bemutatunk környezetfüggetlen és környezetfüggő megoldásokat is a labdarúgó-mérkőzéseken elforduló személynevek felismerésére. Kísérleteink során azt találtuk, hogy az egyszerűbb, környezetfüggetlen szótárbővítés jó választás lehet, ha nem várjuk el a feliratozó rendszertől, hogy az esetragokat is felismerje. Amennyiben pontos ragozással szeretnénk visszaadni a személyneveket, szükség lehet a komplex, környezetfüggő modellezésre.

1 Bevezetés

Televíziós műsorok feliratozása során nagy kihívás jelent az **élő műsorok feldolgozása**. Élő adás esetén a feliratnak is élőben kell születnie, nincs idő az alapos szerkesztésre. A tavalyi Magyar Számítógépes Nyelvészeti (MSZNY) konferencián bemutattunk egy közéleti- és hírműsorok gépi úton történő, élő feliratozásához fejlesztett rendszert, melyet egy a Médiaszolgáltatás-támogató és Vagyonkezelő Alap (MTVA) által támogatott kutatás keretében hoztunk létre [1]. A közéleti- és hírműsorokon túl a legnagyobb közérdeklődére a sportműsorok, azon belül is a **labdarúgó-mérkőzések** számítanak. Kutatásunk célja, hogy ennél a műsortípusnál is ki tudjuk váltani a költséges és gyakran hiányos feliratot eredményező gépelést, automatikus beszédfelismerő rendszerre épülő megoldással.

Az élő feliratozás sosem könnyű feladat, de míg a közéleti- és hírműsorok döntően stúdióban rögzített tervezett és olvasott beszédet tartalmaznak, addig a sportkommentárok tipikusan zajos környezetben születnek és a nehezebben modellezhető spontán beszédhez állnak közelebb. Ezeken felül a legnagyobb problémát az idegen személynevek jelentik. Méréseink szerint a labdarúgó kommentárokban átlagosan **minden hetedik szó egy személynév**, melyek kiejtése sem feltétlenül követi a magyar nyelv szabályait. Kijelenthetjük tehát, hogy a jó minőségű sportfeliratozás egyik kulcsa a személynevek minél pontosabb felismerése.

Cikkünkben több lehetséges módszert is ismertetünk, melyek segítségével egy már betanított beszédfelismerő modellbe utólag is elhelyezhetők személynevek, abban az esetben is, ha az eredeti modell nem tartalmazott rájuk mintát. Elsőként bemutatjuk a legegyszerűbb, **környezetfüggetlen megoldásokat**, melyeknél a nyelvi modellbe önálló szavakként helyezzük el az új személyneveket. Ezután megmutatjuk, hogy hogyan lehetséges az új személyneveket az előfordulásuk lehetséges kontextusa alapján **környezetfüggő módon** modellezni. Mindkét modellezési megközelítést kipróbáltuk alanyesetben álló és esetragokkal ellátott személynevekkel is. Kiértékeléshez egy a 2016-os labdarúgó-Európa-bajnokság alatt rögzített kétórás tesztfelvételt használtunk, melyen a szokásos szóhiba-arányon túl a különböző modellezési módszerekkel elérhető személynév-visszakeresés hatékonyságát (felidézés, pontosság, F1) is mértük.

A következő fejezetben cikkünk témaköréhez legjobban illeszkedő nemzetközi és hazai eredményeket mutatjuk be. Ezután a kísérleti feliratozó rendszer akusztikus modelljének felépítését, valamint tanítóadatbázisait ismertetjük. A negyedik fejezetben mutatjuk be azokat a módszereket és adatokat, melyekkel a rendszer kezdeti nyelvi modelljét tanítottuk, majd részletesen ismertetjük a használt szótárbővítési eljárásokat. A hatodik fejezetben a feliratozó rendszer különböző konfigurációkban mérhető szóhibaarányát és személynév felismerési pontosságát hasonlítjuk össze, míg végül az utolsó fejezetben összefoglalását adjuk vizsgálataink legfontosabb eredményeinek.

2 Kapcsolódó eredmények

A televíziós műsorok feliratozásnál a felirat elkészítésre rendelkezésre álló idő és erőforrások határozzák meg a felirat minőségét [2]. Ha kevés idő áll rendelkezésre (élő feliratozás), jó minőségű felirat csak aránylag nagy ráfordítással hozható létre. Éppen ezért a műsorszolgáltatók már régóta alkalmaznak gépi beszédfelismerést a kézi módszerek mellett, így ugyanis adott feliratozási idő (késleltetés) mellett jobb minőségben (vagy kisebb költséggel) tudnak feliratozni [3, 4]. Sőt a technológia fejlődése mára lehetővé tette, hogy bizonyos műsortípusokat teljesen **gépi úton feliratozzanak**, és csak szükség esetén kapcsoljanak át kézi feliratozásra [2]. Mások inkább a hallottakat elismétlő vagy összefoglaló ún. **újrabeszélők** alkalmazásával találták meg az egyensúlyt a gépi és emberi feldolgozás között [5, 6]. Elvi lehetőségként a műsorfolyam késleltetése is felmerül, hiszen így időt lehet nyerni a felirat elkészítéséhez. Ezt azonban jogi és tartalompolitikai megfontolásból a szolgáltatók nem preferálják.

Korábbi magyar nyelvű sportfeliratozó rendszerről nincs tudomásunk. Magyar nyelvű, gépi feliratozás témakörében is csak tavalyi MSZNY cikkünkre tudunk hivatkozni [1], ahol azonban híradókat és közéleti témájú műsorokat feliratoztunk, melyek

akusztikai és nyelvi szempontból is könnyebb feladatnak számítanak. Szerencsére azonban más nyelveken már készültek sportműsor feliratozó rendszerek. Ezek közül két tanulmányt emelnénk ki, melyek a Sochiban rendezett téli olimpiai játékok cseh [5] illetve orosz [6] nyelvű feliratozását mutatják be. Bár mindkét kutatócsoport használt osztály n-gram alapú **tulajdonnév modellezést**, de ennek a **hatékonyságát nem értékelték ki**, és nem vizsgálták, hogy mire képesek az egyszerűbb, a tanítószöveg előkészítése nélkül is alkalmazható, környezetfüggetlen módszerek.

A fenti két rendszer szóhiba-aránya a konkrét sportágtól függően 25-45% között mozgott, amikor a teljesen automatikus feliratozást értékelték ki. Mivel ez a hibaarány a feliratozáshoz túl magas, mindkét kutatócsoport **újrabeszélőket alkalmazott**. Az újrabeszélés számos szempontból könnyebbé teszi a felismerési feladatot. Egyrészt az újrabeszélők csendes környezetben dolgoznak, hangjukhoz adaptált akusztikus modell ismeri fel a beszédüket és a kommentátorral ellentétben több hónapos képzésen sajátítják el, hogyan tudnak minél érthetőbb feliratokat létrehozni gépi beszédfelismerő segítségével [7]. Az újrabeszélés ugyanis egyben **élő feliratszerkesztést is jelent**, ahol a gyors tempó miatt nehezen kivitelezhető szöveghű ismétlés helyett inkább a tartalmi összefoglaláson van a hangsúly. A cseh rendszerben például a jégkorong-mérkőzések feliratozásánál csak az elhangzott kommentár 63%-át feliratozták, ugyanis az összefoglaláshoz szükséges idő kb. 5 másodperc késleltetést jelentett a feliratban, mely értelmetlenné tette többek között a korongot birtokló játékosok nevének újrabeszélését. Az orosz rendszerben webes adást feliratoztak, így a műsorfolyam késleltetésével kompenzálták a felirat csúszását.

A fentiek összefoglalásaként azt mondhatjuk, hogy az újrabeszéléssel sokat egyszerűsödik a beszédfelismerési feladat, ezért az újrabeszélt feliratok kis hibával rendelkeznek (1-5% szóhiba-arány). Ellenben csak 60-80%-ban hűek az eredeti szöveghez és nagy (minimum 5 másodperc [8]) felirat késleltetéssel járnak. Az újrabeszélés megvalósítása döntően nem technológiai, hanem oktatási és szervezési kérdés. Magyar nyelven összefoglaló jellegű újrabeszélés legjobb tudomásunk szerint egyelőre nincs tervbe véve.

3 Akusztikai modellezés

Ebben a fejezetben az akusztikus modell tanításához felhasznált adatbázisokat, tanítási módszereket valamint a kiejtésmodellezést ismertetjük. Kísérleteink során végig az itt bemutatott akusztikus modellt alkalmaztuk, és csak az ehhez kapcsolódó felismerő hálózatot változtattuk a személynevek modellezéséhez.

3.1 Akusztikai tanító-adatbázisok

Az akusztikus modell tanításához használt adatbázis alapját a tavalyi MSZNY-en bemutatott feliratozó rendszerünkben [1] is használt hanganyagok adták. Ez a közel 500 órás adatbázis webes híradókból, közéleti hírműsorokból, az Egri Katolikus Rádió felvételeiből (EKR), a magyar Speecon [9] adatbázisból és egy félig felügyelten annotált

MTVA adatbázisból (FF) áll. Ezt az adatbázist egészítettük ki további 26 óra M1 csatornáról rögzített hírháttér műsorral (HH). Az egyes műsortípusokhoz tartozó felvételek időtartama az **1. táblázatban** található meg.

1. táblázat: Az akusztikai tanító-adatbázisok mérete

	Webes híradók	Közéleti hírműsorok	FF	EKR	Speecon	HH	∑
Időtartam [óra]	64	31	100	65	228	26	**514**

3.2 Akusztikus modellek tanítása

Az akusztikus modellek tanítása Kaldi keretrendszerben [10] történt, de front-endként a SpeechTex Kft. dekóderének (VOXerver [11]) a lényegkiemelő modulját használtuk. A tanításhoz az 514 órás egyesített szélessávú korpuszunkat használtuk. A tanítás a state-of-the-art-nak megfelelő módon, mel-frekvenciás kepsztrális jellemzőkön (MFCC), trifón Gaussian mixture- és rejtett Markov-modellek (GMM/HMM) készítésével indult. A jellemzővektorokat 13 dimenziós kiindulási MFCC-kből, a front-endben 9 (aktuális ±4 keret) összefűzése után alkalmazott lineáris diszkriminancia-analízis után 40 dimenziós keretekként kaptuk. A tanított GMM/HMM modellek 9730 osztott állapottal, állapotonként átlagosan 10 Gauss-komponenssel rendelkeztek. Az ebből kiindulva készített előrecsatolt **neurális hálózat** bemeneti rétege 360 dimenziós (aktuális keret ±4 keret összefűzve), 6 rejtett rétege 400 egységből, egységenként 5 neuronból állt (összesen 2000 neuron rejtett rétegenként), p-norm aktivációs függvényekkel.

3.3 Kiejtésmodellezés

A lexikai elemek fonetikus átírását szabályalapú, automatikus eljárással készítettük, mely megengedi, hogy a kivételes ejtésű, elsősorban nem magyar szavak átiratát kézzel adjuk meg. A szótárbővítési módszerek tesztelése során minél pontosabban szerettünk volna szimulálni egy valós helyzetet. Éppen ezért nem vettük fel a kivételes ejtésű szavak közé a tesztanyagban előforduló összes névelemet, csak azokat, melyeket megítélésünk szerint minden körültekintő ember felvenne a feliratozás témaköre alapján. Mivel a tesztanyag a labdarúgó-Európa-bajnokság közvetítéseiből lett válogatva, úgy döntöttünk, hogy az **EB-re nevezett játékosok, bírók és szövetségi kapitányok** nevét (összesen 595 név) írjuk át és helyezzük el a listán.

4 Kezdeti nyelvi modell

Kísérleteink kiindulási alapja egy általános sport témakör alapján tanított nyelvi modell volt, melyhez a tanítószövegeket a 2016-os labdarúgó-Európa-bajnokság előtt gyűjtöttük, így biztosítva a tanítás és tesztelés közötti függetlenséget.

4.1 Szöveges tanító-adatbázisok

Kísérleti rendszerünk nyelvi modelljének betanításához három, különböző forrásból származó szövegkorpuszt használtunk fel (lásd **2. táblázat**):

- *Sportműsor leiratok*: Az MTVA által rendelkezésünkre bocsátott sportműsorok közül válogatott adatbázis kézi leiratai. Az adatbázis a 2016-os EB előtt rendezett labdarúgó-mérkőzéseket is tartalmaz.
- *Sporthír feliratok*: Az MTVA csatornáin sugárzott sporthírekhez tartozó feliratok
- *Sport webkorpusz*: A kisebb méretű, de feladatspecifikus tanítószövegek mellett kiegészítő adatbázisként egy tematikus weboldalakról gyűjtött korpuszt is felhasználtunk a kísérleti rendszerben

2. táblázat: A szöveges tanító-adatbázisok statisztikai adatai

	Sportműsor leiratok	Sporthír feliratok	Sport webkorpusz	∑
Token [millió szó]	0,380	0,668	32,2	33,2
Type [ezer szó]	38	66	798	818

4.2 Kezdeti nyelvi modell tanítása

A szövegkorpuszok előkészítése során eltávolítottuk a nem lexikai elemeket (pl. számok, írásjelek, rövidítések), meghatároztuk a mondathatárokat, majd statisztikai módszer segítségével átalakítottuk a mondatkezdő szavakat, oly módon, hogy csak a tulajdonnevek őrizzék meg a nagy kezdőbetűs írásmódot. Ezután bizonyos nem lexikai elemeket átírtunk kiejtett alakjukra (pl. '3', *három*), ezzel segítve a kiejtési modell generálását. A normalizált tanítószövegek alapján minden korpuszon független, 3-gram nyelvi modellt tanítottunk az SRI nyelvi modellező eszköz segítségével [12]. A kezdeti sport feliratozó nyelvi modellje ezután úgy készült, hogy az egyes modelleket lineáris interpoláció segítségével a beszédfelismerési feladathoz adaptáltunk a tesztanyag paraméterhangolási célokra elkülönített részén.

5 Szótárbővítési módszerek

Kísérleteink célja az volt, hogy az általános sport témakörben tanított és az előző fejezetben bemutatott kezdeti nyelvi modell szótárát kibővítsük a labdarúgó EB közvetítésében legnagyobb valószínűséggel elhangzó személynevekkel. A fejezet első felében bemutatjuk a szótárbővítéshez használt névlista összeállításánál alkalmazott módszereket, majd a második felében a szótárbővítési eljárásokat ismertetjük.

5.1 Személynév-listák összeállítása

Az első lista, amit összeállítottunk a szótárbővítéshez használt személynevek **alanyesetben** álló alakjait tartalmazta. Ez a 3.3 fejezetben már bemutatott 595 néven alapult (EB résztvevő játékosok, bírók és szövetségi kapitányok), azonban a teljes neveken túl minden vezetéknév is még egyszer a listára került, mivel a kommentátorok gyakran így hivatkoznak az egyes személyekre (összesen 1190 személynév).

Nem csak alanyesetű személyneveket szerettünk volna azonban visszaadni, így a listán szereplő neveket a korpuszban előforduló leggyakoribb **hat főnévi esetragga**l is elláttuk, ezek gyakorisági sorrendben a következők: részes-, tárgy-, eszközhatározói-, ablativus-, delativus- és allativus esetek. Méréseink szerint ezzel a hat esettel a korpuszban előforduló ragozott főnevek 85%-át tudjuk modellezni. Mivel a magyar nyelvben a ragok egy része – mint a fentebbi hat eset is – váltakozó, tehát magánhangzó-harmónia szerint illeszkednie kell a szótőhöz, a ragok generálása illesztés alapján történt. A szótövek magánhangzó-harmónia osztályba sorolása szabályalapú módszerrel zajlott: a nevek fonetikus átirataiban szereplő magánhangzók alapján. Három osztályt különböztettünk meg: hátulképzett (-hoz), elölképzett kerekített (-höz) és elölképzett kerekítetlen (-hez). A tárgyeset rag változatának kiválasztásakor nem csak a szótő magánhangzó-harmónia osztálya, hanem mássalhangzóra végződés esetén a mássalhangzó fonetikus tulajdonságai is szerepet játszottak. A folyamat végén tehát az alanyesetben álló neveken túl még további hat darab az 1190 név különböző főnévi eseteit tartalmazó lista állt rendelkezésünkre.

5.2 Környezetfüggetlen szótárbővítés

A környezetfüggetlen szótárbővítés lényege, hogy az új szavakat (jelen esetben személyneveket), környezetükből kiragadva, izoláltan helyezzük el a nyelvi modellben.

5.2.1 Elhelyezés a tanítószövegben

Az első szótárbővítési módszer, amit vizsgálunk, az a szélsőségesen egyszerű megoldás, amikor az új szavak listáját egyszerűen felsoroljuk a nyelvi modell tanítószövegében. Bár ezzel minden új szó nullánál nagyobb valószínűséget kap, valódi statisztikai tanítás nem történik, így az új szavak valószínűsége távol fog esni a valós gyakoriságuktól. Ezt a módszert leginkább kíváncsiságból és **csupán az alanyesetben álló névlistával** próbáltuk ki.

5.2.2 Interpoláció unigram modellel

A második környezetfüggetlen szótárbővítési módszer lényege az, hogy az új szavak listájából építünk egy unigram nyelvi modellt uniform valószínűségekkel, és ezt a modellt interpoláljuk a kezdeti rendszerben bemutatott nyelvi modellekkel. Elsőre nagyon hasonlónak tűnik az előző megoldáshoz, de nagy előnye, hogy az új szavak nyelvi modelljéhez tartozó **interpolációs súly** segítségével hangolni tudjuk az új szavak valószínűségét.

Kétféle szótárbővített modellt készítettünk ezzel a módszerrel. Az első esetben az unigram modell az **alanyesetben** lévő 1190 szavas névlista alapján lett összeállítva. A

második esetben nem csak az alanyesetű személyneveket, hanem az 5.1 fejezetben bemutatott módszerrel generált egyéb **ragozott alakokat** is felvettük a modellbe, mely így összesen 8330 szótárelemmel bővült.

5.3 Környezetfüggő szótárbővítés

A környezetfüggő szótárbővítés lényegi gondolata, hogy azonosítjuk a tanítószövegben azokat a szavakat, melyek az új szótári elemekkel azonos szerepet töltenek be. Ezután az új szavak listáját elhelyezzük a nyelvi modellben minden olyan kontextusba, melyet relevánsnak azonosítottuk. Így a környezetfüggetlen modellekkel ellentétben az új szavak a valóságos használatukat jól közelítő kontextussal kerülhetnek be a felismerő hálózatba. Ennek a technikának a lépeseit ismertetjük a következőkben.

5.3.1 Személynevek azonosítása a kézi leiratokban

Elsőként tehát releváns kontextussal rendelkezdő szavakat, azaz a személyneveket kell azonosítanunk a tanítószövegben. A rendelkezésünkre álló kézi leiratokban a személynevek harmada volt kézzel felcímkézve. A korpusz 5%-át, melyben az összes személynevet felcímkéztük, tesztelési célokra félretettük. A korpusz fennmaradó részét három iterációban címkéztük fel. Baseline modellnek azt a naiv osztályozást tekintettük, amikor minden nagy kezdőbetűs szó személynév címkét kapott. Ez alacsony pontosságot (74%) és magas felidézési arányt (100%) jelentett (lásd **3. táblázat**).

3. táblázat: Személynevek szöveges visszakeresésének hatékonysága a sport leiratokban

Modell	Pontosság [%]	Felidézési arány [%]	F1 [%]
Baseline	74	100	84
SzegedNE	94	74	83
Stanford: csak címkézett sorok	76	97	85
Stanford: hiányos címkézés	99	47	64
Stanford-Szeged 1. kör	96	83	88
Stanford-Szeged 2. kör	95	88	91

A statisztikai tanuláson alapuló címkézést két eszköztár segítségével végeztük, a **Stanford-NER** [13] CRF-alapú szekvencia osztályozójával, valamint a **SzegedNE** [14] magyar nyelvre tanított modelljével. Először a Stanford-NER-t tanítottuk a rendelkezésre álló hiányosan címkézett korpusszal. Két tanítást is végeztünk: egyet a teljes hiányosan címkézett korpusszal, egyet pedig a korpusz azon soraival, ahol minden nagy kezdőbetűs szó fel volt címkézve. Ez utóbbi sorokat tartalmazó részkorpusz feltevésünk szerint helyesen címkézettnek tekinthető. A teljes hiányosan címkézett korpuszon tanított modell alacsony pontosságot (76%) és magas felidézési arányt (97%) hozott, a csak felcímkézett sorokon tanított modell pedig magas pontossággal (99%) és alacsony felidézéssel (47%) járt. A cél a kettő közti optimális egyensúly megtalálása volt.

Ezután felcímkéztük a korpuszt a SzegedNE osztályozójával (F1: 83%), majd az új címkéket egyesítettük a hiányos kézi címkézéssel, és az így keletkezett címkézéssel tanított új Stanford-modell már 96%-os pontosságot és lényegesen magasabb 83%-os

felidézést hozott. Utolsó lépésben az addigi legjobb modellel (Stanford-Szeged 1. kör) felcímkézett korpusszal új modellt (Stanford-Szeged 2. kör) tanítottunk, ez **95%-os pontosságot és 88%-os felidézést** jelentett a teszthalmazon. A címkézés és kiértékelés IO (inside-outside) jelölési konvenció szerint történt.

A személynevek azonosítását követően kétféle címkézett tanítószöveg hoztunk létre. Az elsőben az esetragtól függetlenül minden egytagú személynév NAME és minden többtagú COMPOUND címkét kapott. A második címkézésnél a főnévi eseteket is azonosítottuk a Magyarlánc [15] eszköztár segítségével, és jelöltük is a címkékben (pl. NAME-ACC, COMPOUND-DAT).

5.3.2 Címkék helyettesítése a felismerő hálózatban

A feliratozó rendszer mögött dolgozó gépi beszédfelismerő **súlyozott, véges állapotú átalakítókat** (WFST) [16] használ a különböző szintű nyelvi információk integrációjához. Ez a technológia kiválóan alkalmas arra is, hogy a tanítószövegben elhelyezett címkéket a névlistákkal helyettesítsük.

Először a címkékkel ellátott kézi leiratokból osztály n-gram nyelvi modellt tanítottunk, melyben a személynevek helyén az esetrag nélküli vagy esetraggal ellátott címkék álltak. Ezután a címkéket tartalmazó nyelvi modellt interpoláltuk az összes korábban bemutatott nyelvi modellel (lásd 4.2 fejezet). Az immáron személynév címkéket is tartalmazó interpolált nyelvi modellt WFST-vé alakítottuk és az ún. **kompozíció művelettel** [16] egyesítettük a névlistákból alkotott WFST-vel. Az így létrejött WFST már tartalmazta a megfelelő esetben álló személynevek listáját a címkék helyén. Ez a WFST képezi az alapját az új felismerő hálózatnak, mely a névlistákon található személyneveket a főnévi esetüknek megfelelő kontextusban képes modellezni.

6 Eredmények

Ez a fejezet kutatásunk eredményeit foglalja össze. Ismertetjük a kezdeti és szótárbővített modellek feliratozási és személynév-felismerési hatékonyságát, valamint ezek meghatározáshoz használt módszereket.

6.1 Kiértékelés

A feliratozó rendszer tesztelésére az MTVA által sugárzott 2016-os labdarúgó-Európa-bajnokság magyar nyelvű kommentárjait használtuk fel. Egy összesen kétórás adatbázist állítottunk össze különböző nemzetek által vívott mérkőzések véletlenszerűen válogatott részleteinek felhasználásával. Miután elkészítettük a pontos szöveges átiratot, az adatbázist két részre osztottuk: 29 percet félreraktunk a fejlesztés során felmerülő szabad paraméterek hangolásához, 94 percet pedig a végső kiértékeléshez használtunk.

A korábbi fejezetekben ismertetett modellekből minden esetben súlyozott, véges állapotú átalakítókat építettünk, melyekből a SpeechTex VOXerver [11] segítségével készítettük el a tesztanyag feliratait. A feliratozási pontosság megállapításához a beszédfelismerő rendszerek kiértékelésénél leggyakrabban alkalmazott metrikát a **szóhibaarányt** használtuk. A személynevek felismerési hatékonyságának megállapításához

először elkészítettük a vizsgált felirat és a referencia leirat szószintű összerendelését, majd ez alapján kiszámoltuk a **nevek felismerésének pontosságát** és **felidézési arányát**. A kiértékeléshez használt névlistát az 5.3.1 fejezetben ismertetett eljárással nyertük ki a tesztadatbázisból, így az természetesen olyan neveket is tartalmazhatott, melyet nem használtunk a szótárbővítéskor (lásd **4. táblázat**). Csak teljes egyezést fogadtunk el találatnak, azaz például egy hibás raggal felismert név egy fals pozitív és egy fals negatív hibát is eredményezett.

4. táblázat: A tesztadatbázis statisztikai adatai

Teszthalmaz	Hangfelvétel hossza [perc]	Szavak száma	Személynevek száma
Fejlesztő	29	2189	268
Kiértékelő	94	6975	989

6.2 Feliratozási eredmények

A kiértékelés első szakaszában a hagyományos beszédfelismerési metrikákat használtuk a különböző szótárbővítési módszerek vizsgálatához. A perplexitást és a szótáron kívüli szavak arányát a kiértékelő teszthalmazon mértük, és mindkettő a nyelvi modellek szöveges illeszkedéséről szolgáltat információt. A szóhiba-arány a gépi felirat szövegének a referenciához mért hibaarányát mutatja meg.

5. táblázat: A nyelvi modellek és a feliratozás kiértékelése

Szótár-bővítő névlista	Nyelvi modell	Perplexitás [-]	Szótáron kívüli szavak aránya [%]	Szóhiba-arány [%]
-	Kezdeti modell (EB nevek kiejtése nélkül)	630	2.3	33.8
-	Kezdeti modell	630	2.3	32.3
EB névlista	Elhelyezés a tanítósz.-ben	580	2.3	31.3
EB névlista	Unigram interpoláció	534	2.3	29.9
EB névlista	Környezetfüggő m.	**490**	**2.3**	**29.6**
EB névlista +esetragok	Unigram interpoláció	606	1.7	30.5
EB névlista +esetragok	Környezetfüggő m.	**513**	**1.7**	**29.3**

Az **5. táblázat** eredményeit vizsgálva azt láthatjuk, hogy az EB névlistákkal történő szótárbővítés önmagában nem csökkentette a szótáron kívüli szavak arányát, mivel az alanyesetű személynevek többségét már eleve tartalmazta a kezdeti modell. Lényeges csökkenést csak akkor látunk, amikor a ragozott személynevekkel is bővítjük a szótárat, tehát a ragozott alakokból már keveset tartalmazott az eredeti tanítószöveg. Ezzel szemben a **perplexitás határozottan csökken** ahogy egyre komplexebb szótárbővítési technikákat vezetünk be, ami arra utal, hogy ezek a technikák valóban nagyobb pontossággal becsülik a személynevek valószínűségét.

A feliratozás szóhiba-aránya három esetben csökkent számottevően. Az első csökkenés ahhoz köthető, amikor megadjuk az EB névlista szereplőinek kiejtését a modellben. Másodszorra akkor csökken, amikor az EB személyneveket elhelyezzük a tanítószövegben, majd ezután akkor, amikor unigram modell interpolációt is alkalmazunk. A környezetfüggő modellezés jelentős javulást ezen felül már nem hoz. Egészen addig igaz ez, míg nem követeljük meg a személynevek esetragjainak visszaadását. A ragozott személynevek visszaadásakor **a környezetfüggő modellezés egyértelmű előnyre tesz szert**. Feltehetően az az oka ennek, hogy a ragozott személynevekkel feltöltött listák akusztikailag könnyen összetéveszthető szavakat tartalmaznak, így ezek elhelyezésénél a kontextus ismerete felértékelődik.

6.3 Személynév-felismerési eredmények

A személynév-felismerési eredmények úgy készültek, hogy a tesztanyag összes személynevét kigyűjtöttük és információ-visszakeresési problémaként értelmezve a feladatot megvizsgáltuk, hogy milyen hatékonysággal adja vissza őket a gépi felirat. Fontos megjegyezni, hogy a tesztanyag névlistája természetesen eltér a modellbővítéskor használt névlistától, hiszen a tesztanyag nem tartalmazza az összes lehetséges EB személynevet, ellenben sok olyan nevet is tartalmaz, melyeket a bővítőlista nem.

6. táblázat: Személynév felismerési eredmények

Szótárbővítő névlista	Nyelvi modell	Pontosság [%]	Felidézési arány [%]	F1 [%]
-	Kezdeti modell (EB nevek kiejtése nélkül)	90.0	32.0	47.2
-	Kezdeti modell	91.4	39.8	55.5
EB névlista	Elhelyezés a tanítósz.-ben	91.1	44.6	59.9
EB névlista	Unigram interpoláció	**90.8**	**51.9**	**66.0**
EB névlista	Környezetfüggő m.	90.3	51.1	65.3
EB névlista +esetragok	Unigram interpoláció	89.3	48.1	62.6
EB névlista +esetragok	Környezetfüggő m.	**91.6**	**52.7**	**66.9**

A személynév felismerési eredményeket vizsgálva (**6. táblázat**) azt vehetjük észre, hogy a személynévek felismerési **pontossága minden módszer esetén 90%** körül mozog, attól csak kis mértékben tér el. Azaz a feliratozó rendszer csak kb. minden tizedik felismert személynevet helyettesít másik névvel, és még ezek többsége is a főnévi eset meghatározásához köthető hiba.

A felidézési arányok változatosabb képet mutatnak. Látható, hogy a kezdeti rendszer csak minden harmadik nevet ismer fel a tesztanyagban, míg a szótárbővítés után már minden másodikat. A felidézés esetén is elmondható, amit már szóhiba-aránynál is kifejtettünk: az EB személynevek kiejtésének megadása, a nevek elhelyezése a tanítószövegben, majd az unigram interpoláció mind érdemben növeli a detektált nevek arányát. A környezetfüggő szótárbővítés hatására az alanyesetű nevek felidézési aránya kicsit

visszaesik az unigram interpolációhoz képest. Ahogy azonban a szóhiba-arány esetén is megfigyelhettük, ha az esetragokat is modellezzük, újra a környezetfüggő megoldás kerül előnybe. Összességében tehát az **esetragokkal ellátott, környezetfüggő modell a legjobban teljesítő megoldás**.

7 Összefoglalás

Cikkünkben egy labdarúgó-mérkőzések gépi úton történő, élő feliratozásához fejlesztett rendszert mutattunk be. A labdarúgó-közvetítések kommentárjaiban sok személynév hangzik el, melyek modellezése nagy kihívást jelent. A probléma megoldására különböző szótárbővítési eljárásokat használtunk, melyek két csoportra oszthatók: környezetfüggetlen bővítésnek hívtuk, amikor az új személynevek környezet nélkül, izoláltan kerültek a nyelvi modellbe, környezetfüggőnek pedig azt neveztük, amikor a nevek kontextusát is modelleztük.

A rendszer kiértékeléséhez a 2016-os labdarúgó-Európa-bajnokság mérkőzéseinek magyar nyelvű kommentárjaiból állítottunk össze egy tesztadatbázist, melyen a felirat és a személynevek felismerésének pontosságát is mértük. Általános következtetésként azt vonhatjuk le, hogy a környezetfüggetlen szótárbővítés egy egyszerű és hatékony alternatíva abban az esetben, ha csak alanyesetben álló személyneveket akarunk visszaadni. Amennyiben azonban ragozott személyneveket szeretnénk látni a feliratban, a környezetfüggő modellek alkalmazása tűnik jobb választásnak.

A jelenleg elért 29%-os szóhiba-arány nagyjából megegyezik a sport témájú gépi feliratozás nemzetközi szintjével [5, 6], de még nem ajánlható egyértelműen a kézi feliratozás kiváltására. Megítélésünk szerint további feladatspecifikus akusztikus és szöveges tanítóanyag bevonásával már a **közeljövőben elérhető lehet az a feliratminőség, mely részben kiválthatja a gépelést**, de a nagyon zajos közvetítések esetén még ekkor is indokolt lehet újrabeszélők alkalmazása.

A tanító-adatbázisok bővítésén túl, jövőbeli terveink között szerepel, hogy kipróbáljuk a nem magyar személynevek kiejtésének gépi tanuláson alapuló automatikus generálását, a névlisták elemeihez tartozó apriori valószínűségek tanítását és a kézi leiratokon kívüli tanítószövegek felcímkézését is, hogy azok is hozzájárulhassanak a környezetfüggő modellezésben a kontextus tanításához.

Köszönetnyilvánítás

Ezúton is szeretnénk megköszönni a Médiaszolgáltatás-támogató és Vagyonkezelő Alapnak minden segítséget, mellyel munkánkat támogatta. Kutatásunk részben a Pro Progressio alapítvány és a Patimedia (PIAC_13-1-2013-0234) projekt támogatásával készült.

Bibliográfia

1. Tarján, B., Varga, Á., Tobler, Z., Szaszák, G., Fegyó, T., Bordás, C., Mihajlik, P.: Magyar nyelvű, élő közéleti- és hírműsorok gépi feliratozása. XII. Magyar Számítógépes Nyelvészeti Konferencia: MSZNY 2016. pp. 89–99. , Szeged (2016).
2. Renals, S., Simpson, M.N., Bell, P.J., Barrett, J.: Just-in-time prepared captioning for live transmissions. IBC 2016 Conference. p. 27 (9 .)-27 (9 .). Institution of Engineering and Technology (2016).
3. Evans, M.J.: Speech recognition in assisted and live subtitling for television. BBC (2002).
4. Imai, T., Homma, S., Kobayashi, A., Sato, S., Takagi, T., Saitou, K., Hara, S.: Real-Time Closed-Captioning Using Speech Recognition. ABU Technical Committee 2007 Annual Meeting, Doc. pp. 42–43 (2007).
5. Psutka, J. V., Pražák, A., Psutka, J., Radová, V.: Captioning of live TV commentaries from the olympic games in Sochi: Some interesting insights. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 515–522 (2014).
6. Levin, K., Ponomareva, I., Bulusheva, A., Chernykh, G., Medennikov, I., Merkin, N., Prudnikov, A., Tomashenko, N.: Automated closed captioning for Russian live broadcasting. Fifteenth Annual Conference of the International Speech Communication Association. pp. 1438–1442 (2014).
7. Prazák, A., Loose, Z., Psutka, J., Radová, V., Müller, L.: Four-phase Re-speaker Training System. SIGMAP. pp. 217–220 (2011).
8. Ofcom: Measuring live subtitling quality: Results from the first sampling excercise. (2014).
9. Siemund, R., Höge, H., Kunzmann, S., Marasek, K.: SPEECON-speech data for consumer devices. Second Int. Conf. Lang. Resour. Eval. 883–886 (2000).
10. Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Others: The Kaldi speech recognition toolkit. Proc. ASRU (2011).
11. Tarján, B., Mihajlik, P., Balog, A., Fegyó, T.: Evaluation of lexical models for Hungarian Broadcast speech transcription and spoken term detection. 2nd International Conference on Cognitive Infocommunications (CogInfoCom). pp. 1–5. , Budapest, Hungary (2011).
12. Stolcke, A.: SRILM – an extensible language modeling toolkit. Proceedings International Conference on Spoken Language Processing. pp. 901–904. , Denver, US (2002).
13. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling. Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. pp. 363–370 (2005).
14. Szarvas, G., Farkas, R., Kocsor, A.: A multilingual named entity recognition system using boosting and c4. 5 decision tree learning algorithms. International Conference on Discovery Science. pp. 267–278 (2006).
15. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. Proceedings of RANLP. pp. 763–771 (2013).
16. Mohri, M., Pereira, F., Riley, M.: Weighted finite-state transducers in speech recognition. Comput. Speech Lang. 16, 69–88 (2002).

Mély neuronhálóba integrált spektro-temporális jellemzőkinyerési módszer optimalizálása

Kovács György[1,2], Tóth László[2]

[1]Magyar Tudományos Akadémia, Nyelvtudományi Intézet,
Budapest VI., Benczúr utca 33.
[2]MTA-SzTE Mesterséges Intelligencia Kutatócsoport
Szeged Tisza Lajos körút 103,
e-mail:{gykovacs,tothl}@inf.u-szeged.hu

Kivonat Korábbi munkáinkban szignifikánsan javítottuk a fonémafelismerés pontosságát a jellemzőkinyerés mély neuronhálókba történő integrálásával. Az elkészült keretrendszerben azonban maradtak megválaszolandó kérdések, mint például a jellemzőkinyerési lépés paramétereinek optimalizálása, vagy a Δ és $\Delta\Delta$ együtthatók használata. Jelen munkánkban ezen kérdések megválaszolásával foglalkozunk a TIMIT fonémafelismerési valamint az Aurora-4 szófelismerési feladatokon. Először a TIMIT adatbázist felhasználva próbáljuk a jellemzőkinyerési paramétereket javítani, majd a kapott paramétereket a TIMIT és az Aurora-4 adatbázison értékeljük ki. Megmutatjuk, hogy mind a jellemzőkinyerési paraméterek módosítása, mind pedig a Δ és $\Delta\Delta$ együtthatók általunk javasolt felhasználási módja szignifikánsan javítja az elérhető hibaarányokat.
Kulcsszavak: TIMIT, Aurora-4, spektro-temporális jellemzőkinyerés

1. Bevezetés

Korábbi munkáinkban bemutattunk egy keretrendszert, amely a jellemzőkinyerési szakaszt integrálja a neuronhálóba [1]. Itt meghatározott neuronok bementének megválasztásával, súlyainak megfelelő inicializálásával, valamint azáltal, hogy lineáris aktivációs függvényt rendelünk hozzájuk, elértük, hogy az alsó rétegekben elhelyezkedő neuronok valósítsák meg a jellemzőkinyerés lépését, mintegy szűrőként viselkedve. Később ezt a megoldást fejlesztettük tovább egyenirányított mély neuronhálók valamint konvolúció alkalmazásával [2]. A jellemzőkinyerést végző szűrők méretének optimalizálására azonban kevés figyelmet fordítottunk. Ezt a kérdést két úton tudjuk megközelíteni. Egyrészt módosíthatjuk a szűrők méretét azok „technikai" méretén keresztül, azaz a bemenetet változatlanul hagyva, a szűrők mátrixának sor- és oszlopszámát változtatva. Másrészt módosíthatjuk a szűrők méretét pusztán fizikai méretük változtatásával, azaz a bemenet felbontását úgy variálva, hogy egy ugyanannyi sort és oszlopot tartalmazó szűrőmátrix eltérő nagyságú frekvenciatartományt és időintervallumot fedjen le. Annak érdekében, hogy a korábban létrehozott szűrőket továbbra is alkalmazni tudjuk, ez utóbbi megoldás mellett döntöttünk.

Az általunk használt neuronhálók mel skála szerinti sávszűrőket (mel filterbank) használnak bemenetként. Így azt, hogy adott számú sor eltérő nagyságú frekvenciatartományt fedjen le, könnyen elérhetjük a sávszűrők számának módosításával. Az adott számú oszlop által lefedett időintervallum módosítására a keretezés során alkalmazott keretek méretének, valamint a keretek közti lépésköz méretének módosításával is lehetőségünk nyílik. A keretek közti lépésköz hossza általában 10 és 20 ezredmásodperc között mozog [3], ám újabban mások érdekes eredményeket értek el ezen tartományon kívül eső lépésközök vizsgálatával [4], ezért mi is ez utóbbi megoldás mellett döntöttünk. Az ezen vizsgálatokhoz szükséges matematikai formalizmust a 2. fejezetben vezetjük be. Majd a kísérleti eszközök (adatbázisok és neuronhálók – 3. fejezet) leírása után a 4. fejezetben bemutatjuk a paraméterek optimalizálásához elvégzett kísérleteket.

A szűrők méretének változtatása mellett egy másik módszer, amivel a keretrendszer felismerési eredményeit próbáltuk javítani, a delta és gyorsulási ($\Delta\Delta$) együtthatók felhasználása volt. Ezek keretrendszerünkbe integrálásához (ahogy azt majd részletesen látjuk a 5. fejezet) szükség volt arra, hogy átfogalmazzuk az együtthatók kinyerésének problémáját.

A javasolt változtatásokat a TIMIT fonémafelismerési, valamint az Aurora-4 szófelismerési feladatán teszteltük. Az elvégzett tesztek eredményeit a 6. fejezetben ismertetjük, majd a 7. fejezetben konklúziók levonásával és a jövőbeni tervek ismertetésével zárjuk cikkünket.

2. Jelölések és paraméterek

2.1. Konvolúciós paraméterek

Korábbi munkáinkban két fontos változtatást vezettünk be keretrendszerünkbe [2,5]. Egyrészt, ahogy az napjainkban gyakori [6], egyenirányított neuronokat használtunk, ami azt jelenti, hogy a neuronok a rejtett rétegben hagyományos szigmoid aktivációs függényt helyett a következő függvényt valósítják meg: $max(0, x)$. Másrészt, az általunk alkalmazott neuronhálók Vesely és tsai. [7] nyomán időbeli konvolúciót alkalmaznak. Ennek magyarázatához talán az a legegyszerűbb, ha a konvolúciós réteget több különálló rétegként képzeljük el, amelyek osztoznak súlyaikon. Így a súlyok száma nem változik a konvolúcióval, a bemenetek és kimenetek száma viszont úgy viselkedik, mintha több különálló rétegünk lenne. Az R réteget leíró jellemzők tehát a következők:

- $\mathbf{R^I}, \mathbf{R^O}$: $\mathbf{R}$ réteg be- és kimenete
- $\mathbf{R}^{\#_\mathbf{n}}$: az $\mathbf{R}$ rétegbeli neuronok száma
- $\mathbf{R}^{\mathbf{W_i}}$: az $\mathbf{R}$ réteg i-edik neuronjához tartozó súlyvektor ($0 < i \leq \mathbf{R}^{\#_\mathbf{n}}$)
- $\mathbf{R_{C_m}}$: Ha az $\mathbf{R}$ réteg konvolúciót használ az időtartományban (azaz $\mathbf{R}$ konvolúciós réteg), és bemenetetét X eltérő időpontból veszi ($1 \leq m \leq X$), $\mathbf{R}$ rétegre úgy tekinthetünk, mint X darab különböző rétegre. Ebben az esetben $\mathbf{R_{C_m}}$ jelöli az m-edik ilyen réteget, melynek bemenete $\mathbf{R^I_{C_m}}$, és kimenete $\mathbf{R^O_{C_m}}$ (mivel a súlyokat ezek a rétegek megosztják egymás közt, az X réteg mindegyikének súlyai továbbra is $\mathbf{R^W}$ jelölést kapnak, és az X azonos méretű réteg neuronszámára továbbra is az $\mathbf{R}^{\#_\mathbf{n}}$ jelöléssel hivatkozunk).

2.2. Spektrogram- és ablak-paraméterek

A mel-skálás spektrális ábrázolás vagy röviden spektrogram (S) létrehozásánál többek között az alábbi paraméterek játszanak fontos szerepet:

- $\mathbf{S}^{\delta}_{\mathbf{W}}$: az S létrehozásához használt keret mérete (ezredmásodpercben)
- $\mathbf{S}^{\nu}_{\mathbf{W}}$: az egyes keretek kezdőpontja közti időintervallum (ezredmásodpercben)
- $\mathbf{S}^{\#}_{\mathbf{F}}$: a szűrőkészlet szűrőinek száma

A szűrőkhöz használt ablakok (P - Patch) paraméterei a következők:

- $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{t}}$: $\mathbf{P}$ fizikai mérete az időtartományban. Azt jelzi, hány ezredmásodpercet fed le $\mathbf{P}$.
- $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{f}}$: $\mathbf{P}$ fizikai mérete a frekvenciatartományban.
- $\mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{t}}$: $\mathbf{P}$ „technikai" mérete az időtartományban. Azt jelzi, hány keretet fed le $\mathbf{P}$ (megjegyezzük, hogy ez meghatározható az $\mathbf{S}^{\nu}_{\mathbf{W}}$, $\mathbf{S}^{\delta}_{\mathbf{W}}$ és a $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{t}}$ paraméterekből, ahogy $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{t}}$ is meghatározható az $\mathbf{S}^{\nu}_{\mathbf{W}}$, $\mathbf{S}^{\delta}_{\mathbf{W}}$ és a $\mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{t}}$ paraméterekből).
- $\mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{f}}$: $\mathbf{P}$ „technikai" mérete a frekvenciatartományban. Azt jelzi, hány melszűrőt fed le $\mathbf{P}$ (megjegyezzük, hogy ez meghatározható $\mathbf{S}^{\#}_{\mathbf{F}}$ és $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{f}}$ paraméterekből, ahogy $\mathbf{P}^{\delta_{\mathbf{P}}}_{\mathbf{f}}$ is meghatározható $\mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{f}}$ és $\mathbf{S}^{\#}_{\mathbf{F}}$ paraméterekből).
- $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{f}}$: a szomszédos ablakok átfedési aránya a frekvenciatartományban
- $\mathbf{P}^{\#}_{\mathbf{f}}$: az ablakok száma, amelyek szükségesek a használt frekvenciatartomány lefedéséhez (feltételezve, hogy a szomszédos ablakok átfedése: $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{f}}$)
- $\overline{\mathbf{P}}$: $\mathbf{P}$ (ami egy 2-dimenziós mátrix) vektor formában felírva
- $\overline{\mathbf{P}}^{\delta_{\mathbf{t}}}$: $\overline{\mathbf{P}}$ vektor hossza. Kiszámítható az alábbi formulával: $\mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{t}} \cdot \mathbf{P}^{\delta_{\mathbf{t}}}_{\mathbf{f}}$

Mivel a konvolúciót az időtartományban alkalmazzuk, így neuronhálónk nem csak a frekvencia- de az időtartományban is több ablakot használ bemenetként. Az ehhez kapcsolódó paraméterek és jelölések a következők:

- $\delta_{\mathbf{T}}$: az időtartam (ezredmásodpercben) amit le kívánunk fedni az egymást fedő ablakok használatával
- $\mathbf{P}^{\#_{\mathbf{i}}}_{\mathbf{t}}$: az ablakok száma, amelyek szükségesek a megadott $\delta_{\mathbf{T}}$ időtartam lefedéséhez (feltételezve, hogy közvetlen szomszédos ablakokat használunk)
- $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{t}}$: a szomszédos ablakok átfedési aránya az időtartományban
- $\mathbf{P}^{\#}_{\mathbf{t}}$: az ablakok száma, amelyek szükségesek a megadott $\delta_{\mathbf{T}}$ időtartam lefedéséhez (feltételezve, hogy a szomszédos ablakok átfedése: $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{t}}$)
- $\mathbf{P}^{\nu_{\mathbf{t}}}_{\mathbf{t}}$: azon közvetlenül szomszédos ablakok száma, melyeket kihagyunk, hogy $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{t}}$-nek megfelelő átfedést érjünk el a az ablakok között (ahogy a jelölés sugallja, ez a „technikai" oldala az átfedésnek, amit keretekben adunk meg, míg a fizikai oldala – $\mathbf{P}^{\nu_{\mathbf{P}}}_{\mathbf{t}}$ – értelmezhető ezredmásodpercekben is)

Ha a rendszer által egy adott időpillanatban használt ablakok számát egy mátrixként fogjuk fel, ahol az azonos időtartamból és frekvenciatartományból származó ablakok alkotják a mátrix oszlopait, és sorait, az ablakot amely a mátrix i-edik sorából ($0 < i \leq \mathbf{P}^{\#}_{\mathbf{f}}$) és a j-edik oszlopából ($0 < j \leq \mathbf{P}^{\#_{\mathbf{i}}}_{\mathbf{t}}$) származik jelölhetjuk $\mathbf{P_{ij}}$-vel. Ekkor a korábban bemutatott $\overline{\mathbf{P}}^{\delta_{\mathbf{t}}}$ paraméter a következő jelöléssel kellene rendelkezzen: $\overline{\mathbf{P}}^{\delta_{\mathbf{t}}}_{\mathbf{ij}}$. Mivel azonban kísérleteinkben a különböző idő- és frekvenciatartományból vett ablakok „technikai" mérete megegyezik, a továbbiakban maradunk a korábban bevezetett jelölésnél.

3. Kísérleti eszközök

3.1. TIMIT

A paraméterek hangolásához a TIMIT beszédadatbázist használtuk [8]. A neuronhálók súlyait a 3969 mondatból álló tanítóhalmaz kilencven százalékán tanítottuk, a fennmaradó tíz szálékot pedig a megállási feltétel kiértékelésére, és a paraméterek kiválasztására használtuk. Ezen paraméterek használatával azt követően újabb neuronhálókat tanítottunk, amelyeket a 192 mondatot tartalmazó „mag" (core) teszthalmazon értékeltünk ki. Kiértékelés előtt a fonémacímkéket 39 kategóriába vontuk össze, a bevett gyakorlatnak megfelelően [9].

3.2. Aurora-4

Az Aurora-4 a Wall Street Journal beszédadatbázis zajosított változata [10]. Két 7138 mondatból álló tanítóhalmazt, és 14, egyenként 330 mondatból álló teszthalmazt tartalmaz. A tanítóhalmaz első (tiszta) változata a mondatok zaj nélküli változatát tartalmazza Sennheiser mikrofonnal rögzítve, míg a második változatban az egyes mondatok különböző zajokkal szennyezettek, illetve rögzítésük eltérő mikrofonnal történt. Jelen cikkünkben csak a második (multi-condition) tanítóhalmazt használtuk. Ennek kilencven százalékán tanítottuk a neuronhálók súlyait, míg a fennmaradó részt a megállási feltétel kiértékelésére használtuk.

A kiértékelést az összes teszthalmazon végeztük. Ezen teszthalmazok ugyanazt a 330 mondatot tartalmazzák különböző verziókban: az első hét teszthalmazban lévő hangfájlok rögzítése a Sennheiser mikrofonnal történt, míg a második hét teszthalmazban ettől eltérő mikrofonnal rögzített felvételeket találunk. Mindkét csoport belső felosztása azonos: az első halmaz tiszta beszédet tartalmaz, míg a következő hatban hat különböző zajjal szennyezett beszéd található.

3.3. Neuronháló

A neuronhálók ismertetésének egyszerűsítése céljából leírásukat a különböző funkciókat ellátó rétegek leírására bontjuk.

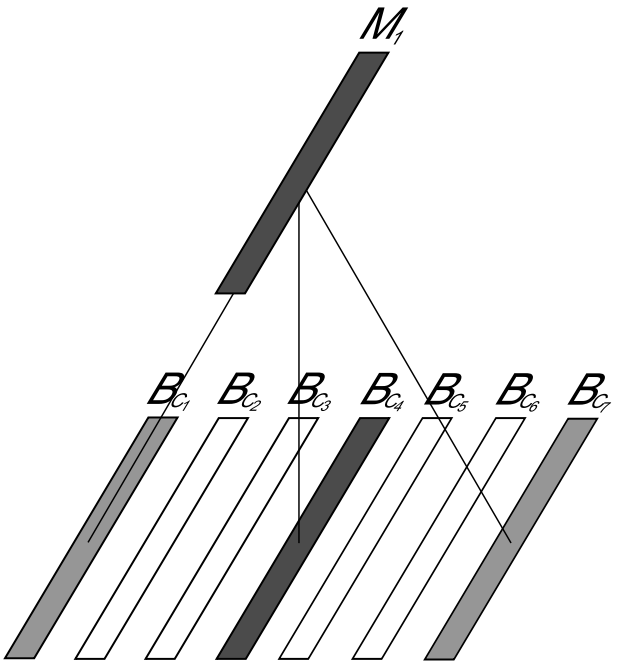
Szűrőrétegek. Minden, a szűrés (Filtering) megvalósításáért felelős réteg ($\mathbf{F_i}$) egy megadott frekvenciatartományból veszi az input-ablakokat. A réteg be- és kimenete a következőképp írható le:

$$\mathbf{F^{I}_{iC_j}} = \overline{\mathbf{P_{ij}}},\ ahol \quad \begin{matrix} 0 < i \leq \mathbf{P^{\#}_{f}} \\ 0 < j \leq \mathbf{P^{\#_i}_{t}} \end{matrix} \tag{1}$$

$$\mathbf{F^{O}_{iC_j}} = [o_{1j}, ..., o_{\mathbf{F^{\#_n}_i}\,j}],$$

$$o_{kj} = \sum_{x=1}^{\overline{\mathbf{P}}^{\delta_t}} \overline{\mathbf{P_{ij}}}[x] \cdot \mathbf{F^{W_k}_{i}}[x] + b_k, \quad ahol\ 0 < k \leq \mathbf{F^{\#_n}_{i}}\,^1 \tag{2}$$

[1] Mivel kísérleteink során minden szűrőréteg ugyanannyi neuront tartalmaz, erre az értékre a későbbiekben a $\mathbf{F}^{\#_n}$ szimbólummal fogunk hivatkozni



1. ábra. A konvolúciós bottleneck réteg ($\mathbf{B}$), valamint az első egyszerű egyenirányított réteg ($\mathbf{M_1}$) illusztrációja. Azt az esetet mutatja be, ahol $\mathbf{P_t^{\#_i}} = 7$, és $\mathbf{P_t^{\nu_t}} = 2$. Egyben illusztrálja azt a tényt is, hogy ablakok megfelelő átfedéséről az időtartományban a bottleneck réteget követő réteg gondoskodik azáltal, hogy a bottleneck réteg megadott számú konvolúciós kimenetét átugorja bemenete beolvasása közben.

„Konvolúciós" réteg. A szűrők megvalósításáért felelős réteget egy (vagy több) további konvolúciós réteg követi, melyek közül az utolsó az „üvegnyak" (bottleneck) réteg. Számunkra itt az első réteg bemenete érdekes:

$$\mathbf{Conv^I_{C_j}} = [\mathbf{F^O_{1C_j}}, ..., \mathbf{F^O_{P_f^{\#}C_j}}],\ ahol\ \ 0 < j \leq \mathbf{P_t^{\#_i}} \tag{3}$$

Egyszerű egyenirányított réteg. Az első nem-konvolúciós egyenirányított réteg ($\mathbf{M_1}$ – lásd 1. ábra) kombinálja a konvolúciós bottleneck ($\mathbf{B}$) réteg kimeneteit, azáltal, hogy kimenetét a következő módon használja fel bemenetként:

$$\mathbf{M_1^I} = [\mathbf{B^O_{C_1}}, \mathbf{B^O_{C_{1+P_t^{\nu_t}+1}}}, ..., \mathbf{B^O_{C_j}}],\ ahol\ \ j = \mathbf{P_t^{\#_i}} \tag{4}$$

Innentől a konvolúciós mély neuronháló (beleértve az $\mathbf{O}$ kimeneti réteget is) ugyan úgy működik, mint bámely hagyományos mély háló, így a további rétegek részletes leírásától eltekintünk.

1. táblázat. Paraméterbeállítások a frekvenciatartományra vonatkozóan

paraméterek	F1	F2	F3	F4	F5
$\mathbf{S}_{\mathbf{F}}^{\#}$	18	26	34	42	50
$\mathbf{P}_{\mathbf{f}}^{\delta_{\mathbf{P}}}$	~1420 mel	~980 mel	~750 mel	~610 mel	~510 mel
$\mathbf{P}_{\mathbf{f}}^{\#}$	3	5	7	9	11

4. Paraméterek optimalizálása

A kísérletek célja az volt, hogy megvizsgáljuk, milyen hatással van a szűrők fizikai méretének változása (a „technikai" méret változtatása nélkül) a felismerési eredményekre. Ezen kísérletekhez bizonyos paramétereket rögzítettnek vettünk:

- $\mathbf{S}_{\mathbf{W}}^{\delta} = 25$ ms
- $\delta_{\mathbf{T}} \approx 265$ ms
- $\overline{\mathbf{P}}^{\delta_{\mathbf{t}}} = 81$ ($\mathbf{P}_{\mathbf{t}}^{\delta_{\mathbf{t}}} = 9$, $\mathbf{P}_{\mathbf{f}}^{\delta_{\mathbf{t}}} = 9$)
- $\mathbf{P}_{\mathbf{t}}^{\nu_{\mathbf{P}}} = \frac{\left\lceil \mathbf{P}_{\mathbf{t}}^{\delta_{\mathbf{t}}}/2 \right\rceil}{\mathbf{P}_{\mathbf{t}}^{\delta_{\mathbf{t}}}}$ ($\mathbf{P}_{\mathbf{t}}^{\delta_{\mathbf{t}}} = 9$ felhasználásával adódik, hogy $\mathbf{P}_{\mathbf{t}}^{\nu_{\mathbf{t}}} = 3$ keret)
- $\mathbf{P}_{\mathbf{f}}^{\nu_{\mathbf{P}}} = \frac{\left\lceil \mathbf{P}_{\mathbf{f}}^{\delta_{\mathbf{t}}}/2 \right\rceil}{\mathbf{P}_{\mathbf{f}}^{\delta_{\mathbf{t}}}}$.

A különböző paraméter-beállítások hatásának tanulmányozására 5 beállítást hoztunk létre a frekvenciatartományra vonatkozó paraméterekre nézve (leolvashatók a 1. táblázatból), és 3 beállítást az időtartományra vonatkozó paraméterekre nézve (leolvashatók a 2. táblázatból). Így összesen 15 párt vizsgáltunk a kísérleteink során.

A vizsgálatához használt neuronhálók bizonyos paraméterei a táblázatokban leírt paraméterek függvényében alakultak, míg mások kötöttek voltak. Ez utóbbiak a következők:

- $\mathbf{F}^{\#_{\mathbf{n}}} = 9$
- $\mathbf{B}^{\#_{\mathbf{n}}} = 200$
- $\mathbf{M}_{\mathbf{1}}^{\#_{\mathbf{n}}} = \mathbf{M}_{\mathbf{2}}^{\#_{\mathbf{n}}} = 1000$
- $\mathbf{O}^{\#_{\mathbf{n}}} = 183/61$ (a három- és egyállapotú fonémamodellekhez)

A paraméterek optimalizálásáért végzett kísérletek során használt neuronhálóknak a szűrők megvalósításáért felelős ($\mathbf{F_i}$) rétegein kívül egyetlen konvolúciós rétegük volt, a bottleneck ($\mathbf{B}$) réteg.

2. táblázat. Paraméterbeállítások az időtartományra vonatkozóan

paraméterek	T1	T2	T3
$\mathbf{S}_{\mathbf{W}}^{\nu}$	10 ms	8 ms	6 ms
$\mathbf{P}_{\mathbf{t}}^{\delta_{\mathbf{P}}}$	~105 ms	~89 ms	~73 ms
$\mathbf{P}_{\mathbf{t}}^{\#_{\mathbf{i}}}$	17	25	33
$\mathbf{P}_{\mathbf{t}}^{\#}$	5	7	11

3. táblázat. Fonémafelismerési hibaarányok (10 függetlenül tanított neuronháló eredményeinek átlaga) a TIMIT validációs halmazán egy- és háromállapotú fonémamodellek esetére (a legjobb eredmények, és az azoktól szignifikánsan nem eltérő eredmények vastagon szedve mindkét esetben).

	61 egyállapotú modell			61 háromállapotú modell		
	T1	T2	T3	T1	T2	T3
F1	21,36%	20,78%	20,68%	21,29%	20,39%	20,04%
F2	*20,65%*	20,20%	20,12%	*20,26%*	19,64%	19,28%
F3	20,18%	19,81%	20,00%	20,05%	19,32%	19,03%
F4	19,89%	**19,65%**	19,79%	19,61%	19,15%	**18,88%**
F5	19,85%	**19,52%**	19,84%	19,64%	19,08%	**18,86%**

4.1. Eredmények

A tanítóhalmazból lehasított tíz százalékra, mint validációs halmazra, megvizsgáltuk a fonémafelismerési eredményeket abban az esetben, ha egyállapotú vagy háromállapotú fonémamodellek segítségével végeztük a tanítást.

Az így kapott eredmények leolvashatók a 3. táblázatból. Korábbi kísérleteink során az F2/T1 paramétereinek megfelelő beállításokat használtuk (az ehhez kapcsolódó beállításokat dőlt betűkkel emeltük ki a táblázatban). Látható, hogy a két esetben a legjobb eredményt adó beállítások eltérnek egymástól. Az abszolút értékben legjobb eredményeket a táblázat jobb oldalán található F4/T3 és F5/T3 beállításokkal kaptuk. Azaz a legjobb felismerési eredményt akkor értük el, ha a keretek közti lépésközt 10 ezredmásodpercről 6 ezredmásodpercre csökkentettük, és a mel-szűrők számát 26-ról 42-re vagy 46-ra növeltük. E két beállítás közül választottunk az előbbit, mivel a használatukkal kapott felismerési eredménye között szignifikáns különbséget nem találtunk, így a szűrők számának további növelését nem láttuk indokoltnak. A 6. fejezetben végzett tesztek során tehát ezt a paraméterbeállítást (F4/T3) fogjuk összehasonlítani az eredeti (F2/T1) beállításokkal, azt vizsgálandó, hogy a javasolt változtatások valóban jobb eredményre vezetnek-e.

5. Delta és gyorsulási együtthatók

Korábbi publikációinkban többször előkerült, hogy a Δ és $\Delta\Delta$ együtthatók hozzáadása a keretrendszerünkhöz hasznos lenne [1,5]. Ezen véleményünket arra alapoztuk, hogy korábban a delta és gyorsulási együtthatók hozzáadása a jellemzőkészlethez javította az elért eredményeinket [11]. A megadott együtthatók használata a jelen keretrendszerben kivitelezhető lenne oly módon, hogy új neuronokat veszünk fel a jellemzőkinyerési réteg után melyek Δ és $\Delta\Delta$ együtthatók kinyerését valósítják meg, és megoldjuk, hogy a hibavisszaterjesztés áthaladjon ezeken a neuronokon. Van azonban egy egyszerűbben kivitelezhető megoldás. A

Δ együtthatók a következő formulával állnak elő:

$$d_T = \frac{\sum_{\theta=1}^{\Theta} \theta(c_{T+\theta} - c_{T-\theta})}{2 \cdot \sum_{\theta=1}^{\Theta} \theta^2}, \tag{5}$$

ahol d_T a Δ együttható T időpontban, a $c_{T-\theta}$ és $c_{T+\theta}$ közötti konstans együtthatókból számítva [12]. Ha ezt a formulát a szűrők alkalmazásával kapott jellemzőkre alkalmazzuk,

$$c_T = \sum_{t=1}^{N} \sum_{f=1}^{M} P_T(f,t) \cdot F(f,t), \tag{6}$$

ahol P mintákat az S spektrogramból a következő módon nyerjük ki:

$$P_T(f,t) = S(f, T+t), \tag{7}$$

a következő egyenletet kapjuk:

$$d_T = \frac{\sum_{\theta=1}^{\Theta} \theta \left(\sum_{t=1}^{N} \sum_{f=1}^{M} F(f,t) \cdot \left(S(f,T+\theta+t) - S(f,T-\theta+t) \right) \right)}{2 \cdot \sum_{\theta=1}^{\Theta} \theta^2}, \tag{8}$$

Mivel az osztó Θ megválasztása után nem függ egyéb paramétertől, adott Θ esetén konstans. Vezessük be tehát a következő konstanst: $\vartheta = 2 \cdot \sum_{\theta=1}^{\Theta} \theta^2$. Ezt használva, valamint újrarendezve (8) egyenletet, a következő formulát kapjuk a Δ együttható számítására, egy jellemző esetén, amit egy S spektrogramra alkalmazott szűrő kimenetként kapunk:

$$d_T = \sum_{t=1}^{N} \sum_{f=1}^{M} F(f,t) \cdot \frac{\sum_{\theta=1}^{\Theta} \theta \left(S(f,T+\theta+t) - S(f,T-\theta+t) \right)}{\vartheta}, \tag{9}$$

Másrészről, ha először alkalmazzuk a (5) egyenletet az S spektrogramra, a spektrogramnak egy Δ változatát kapjuk, ahol az f frekvenciához és t időponthoz tartozó elemet a következőképp kapjuk:

$$S_\Delta(f,t) = \frac{\sum_{\theta=1}^{\Theta} \theta \left(S(f,t+\theta) - S(f,t-\theta) \right)}{\vartheta} \tag{10}$$

Ekkor F szűrő alkalmazása a spektrogramból T időpontban kinyert mintára a következő egyenlet megoldását jelentené:

$$c_T = \sum_{f=1}^{N} \sum_{t=1}^{M} F(f,t) \cdot S_\Delta(f, T+t), \tag{11}$$

ami megegyezik (9) egyenlettel. Tehát ha a célunk egy jellemző Δ együtthatójának kinyerése, azt elérhetjük úgy is, hogy először a spektrogram Δ változatát állítjuk elő, majd abból nyerjük ki a jellemzőt.

6. Kísérleti eredmények

A paraméterek meghatározásához végzett kísérletekkel szemben az alább ismertetett kísérletekben a hálók egymáshoz viszonyított teljesítményén kívül az abszolút teljesítmény is fontos volt. Ezért ezekben a kísérletekben nagyobb hálókat alkalmaztunk. A bottleneck réteg elé további három (egyenként 1000 neuront tartalmazó) konvolúciós réteget helyeztünk el, és az így kapott hálókat két lépésben tanítottuk. Az első lépésben konvolúció nélkül tanítottuk a hálót oly módon, hogy a kimeneti réteget közvetlenül a bottleneck réteg után helyeztük el, majd a következő lépés előtt ezt a kimeneti réteget töröltük, két (egyenként 1000 neuront tartalmazó) réteget és egy új kimeneti réteget vettünk fel, majd egy újabb tanítást indítottunk, ezúttal konvolúció használatával. További módosítás a korábbi kísérletekhez képest, hogy az eredeti beállításaink (F2/T1) esetén a bottleneck réteg 220 neuront tartalmazott (szemben a korábbi 200-al), és a korábban 1000 neuront tartalmazó rétegek neuronszámát 1100-ra növeltük. Ezzel azt biztosítottuk, hogy a különböző jellemzőkinyerési paraméterbeállításokkal dolgozó neuronhálók mérete közel azonos legyen.

6.1. Eredmények a TIMIT beszédadatbázison

A mások által elért eredményekkel való jobb összehasonlítás érdekében ezekben a kísérletekben a tanítást ([6] nyomán) 858 állapot felhasználásával végeztük. A kiértékelés előtt ugyanúgy elvégeztük a 39 kategóriába való összevonást, mint korábban. Az így kapott eredmények leolvashatók a 4. táblázatból. Először vizsgáljuk meg az első két sort, azaz azt a két esetet, amikor a hálók szűrés megvalósításért felelős rétegeit véletlen súlyokkal (1. sor), illetve a korábban bemutatott Gábor szűrők [5] alapján (2. sor) inicializáljuk. Láthatjuk, hogy a súlyok Gábor szűrők alapján történő inicializálása 0,2 százalékpontos hibaarány-csökkenéshez vezet az eredményekben, ami szignifikáns ugyan ($p = 0,025$ értéken), ám minimális. Ezért a további kísérletekben a szűrést megvalósító rétegek súlyait, a háló többi súlyához hasonlóan, véletlen számokkal inicializáltuk. Kiolvasható továbbá a táblázatból, hogy az új paraméterek használatával jelentős javulást értünk el a hibaarányt tekintve (22 százalékos relatív hibacsökkenés), és a Δ valamint $\Delta\Delta$ együtthatók hozzáadásával ezen az eredményen is javítani tudtunk.

4. táblázat. Fonéma szintű hibaarányok (10 függetlenül tanított neuronháló eredményeinek átlaga) a TIMIT „mag" teszthalmazán (a legjobb eredmények, és az azoktól szignifikánsan nem eltérő eredmények vastagon szedve).

Inicializálás	Paraméterek		Δ	PER
	Frekvencia	Idő	$\Delta\Delta$	
Random	F2	T1		24,4%
Gábor	F2	T1		24,2%
Random	F4	T3		18,8%
Random	F4	T3	✓	**18,5%**

5. táblázat. Fonéma szintű hibaarányok (PER) a TIMIT „mag” teszthalmazán (a legjobb eredmények vastagon szedve).

Módszer	PER
Plahl és tsai. [15]	19,1%
Tóth [6]	18,7%
Jelen cikk	*18,5%*
Graves és tsai. [13]	17,7%
Tóth [14]	**16,7%**

Az elért eredményeket összevetve az irodalomban találtakkal (5. táblázat) azt látjuk, hogy a javasolt változtatásokkal a rendszerünk versenyképes eredményeket produkál. Bár az elért fonémafelismerési eredmények elmaradnak Graves és tsai. [13] eredményeitől, ám ők kísérleteikben rekurrens hálókat alkalmaztak. Eredményeink továbbá jelentősen elmaradnak Tóth 2014-es eredményeitől [14], azonban az általa használt hálók az időtartományban *és* a frekvenciatartományban is alkalmaztak konvolúciót, továbbá a dropout módszert is felhasználták. Rendszerünk leginkább ugyanazon szerző egy korábbi cikkében bemutatott rendszerével összehasonlítható [6], melynek eredményein kis mértékben javítani is tudtunk, úgy, hogy az általunk használt hálók méretei csupán negyede az említett cikkben használt háló méretének.

6.2. Eredmények az Aurora-4 beszédadatbázison

Annak érdekében, hogy a neuronhálók teljesítményét különböző zajtípusok (illetve átviteli karakterisztikák) esetén is vizsgálni tudjuk, az elvégzett kísérleteket megismételtük az Aurora-4 szófelismerési feladatára is (a multi-condition tanítóhalmaz felhasználásával). Az eredmények leolvashatók a 6. táblázatból. Ahogy látható, mindkét javasolt módosítás a szófelismerési pontosság javulásához vezetett az Aurora-4 teszthalmazain. A frekvencia- és időparaméterek módosításával 4 százalékos, a Δ valamint $\Delta\Delta$ együtthatók felvételével pedig további 2 százalékos relatív hibaarány-csökkenést értünk el. A különbség szignifikáns mind az első ($p = 0,00005$ értéken) mind pedig a második módosítás esetén ($p = 0,00044$ értéken).

6. táblázat. Szószintű hibaarányok (5 függetlenül tanított neuronháló eredményeinek átlaga) az Aurora-4 teszthalmazán (a legjobb eredmények, es az azoktól szignifikánsan nem eltérő eredmények vastagon szedve).

Inicializálás	Paraméterek		Δ	WER
	Frekvencia	Idő	$\Delta\Delta$	
Random	F2	T1		12,4%
Random	F4	T3		11,9%
Random	F4	T3	✓	**11,6%**

7. táblázat. Szószintű hibaarányok (WER) az Aurora-4 teszthalmazán (a legjobb eredmények vastagon szedve).

Módszer	WER
Chang, Morgan [16]	16,6%
Castro és tsai. [17]	12,3%
D. Baby és tsai. [18]	11,9%
Jelen cikk	***11,6%***

Az eredményeinket az irodalomban talált eredményekkel ismét egy külön táblázatban (7. táblázat) hasonlítjuk össze. Chang és Morgan [16] hozzánk hasonlóan mély konvolúciós hálókat alkalmaztak, melyek alsó rétegébe szűrők együtthatóit építették be, ám velünk ellentétben náluk bemenetként a PNS (Power Normalized Spectrum) szolgált, továbbá ők több és nagyobb szűrőket alkalmaztak, de nem használták a Δ valamint gyorsulási együtthatókat. Castro és tsai. [17] szintén felhasználtak Gábor szűrőket is, ám legjobb eredményeiket az úgynevezett Amplitude Modulation Filter Bank (AMFB) segítségével érték el. Valamint szintén mély neuronhálókat alkalmaztak, ám ők ezt a beépített Kaldi recept alapján, előtanítás használatával tették. További különbség, hogy a mieinknél jelentősen nagyobb (7 rejtett rétegű, rétegenként 2048 neuront tartalmazó) hálókat használtak. D. Baby és tsai. [18] egy a miénktől jelentősen eltérő megközelítést a minta-alapú beszédkiemelés módszerét alkalmazták, hozzánk hasonlóan egy DNN/HMM hibrid architektúrába (ám Castrohoz és társaihoz hasonlóan a mieinknél jelentősen nagyobb – 6, egyenként 2048 rétegből álló – neuronhálót használva). Ahogy a 7. táblázatból látható, az általunk elért legjobb eredmények felülmúlják a három említett cikkben bemutatott eredményeket (30 és 2,5 százalék közötti relatív hibaarány-csökkenéssel).

7. Konklúzió és jövőbeni munka

Cikkünkben két módosítást javasoltunk az általunk használt keretrendszerhez, a TIMIT beszédadatbázison végzett kísérletek, valamint korábbi kísérleteink alapján. A TIMIT fonémafelismerési, valamint az Aurora-4 szófelismerési feladaton végzet kísérletek nyomán azt láttuk, hogy mindkét módosítás az eredmények szignifikáns javulásához vezet. Bár a TIMIT adatbázison elért fonémafelismerési eredményeken azt láttuk, hogy a háló jellemzőkinyerésért felelős rétegeiben található súlyok Gábor-szűrők alapján történő inicializálása szignifikáns javulást eredményezett, ez a javulás minimális volt. A későbbiekben érdemes lehet megvizsgálni a két lépéses tanítás hatását a szűrőkre (beleértve azt az esetet, amikor az első vagy második lépés során a szűrőkhöz kapcsolódó súlyokat változatlanul hagyjuk) valamint a kimenetként kapott szűrők hasznosságára a felismerési eredmények szempontjából.

Hivatkozások

1. Kovács, Gy., Tóth, L.: The joint optimization of spectro-temporal features and neural net classifiers. In: Proc. TSD. (2013) 552–559
2. Kovács, Gy., Tóth, L.: Joint optimization of spectro-temporal features and deep neural nets for robust automatic speech recognition. Acta Cybernetica **22**(1) (2015) 117–134
3. Picone, J.W.: Signal modeling techniques in speech recognition. Proceedings of the IEEE **81**(9) (1993) 1215–1247
4. Pundak, G., Sainath, T.: Lower frame rate neural network acoustic models. In: proc. Interspeech. (2016) 22–26
5. Kovács, Gy., Tóth, L., Van Compernolle, D.: Selection and enhancement of Gabor filters for automatic speech recognition. International Journal of Speech Technology **18**(1) (2015) 1–16
6. Tóth, L.: Convolutional deep rectifier neural nets for phone recognition. In: Proc. Interspeech, IEEE (2013) 1722–1726
7. Veselý, K., Karafiát, M., Grézl, F.: Convolutive bottleneck network features for LVCSR. In: Proc. ASRU. (2011) 42 – 47
8. Lamel, L., Kassel, R., Seneff, S.: Speech database development: Design and analysis of the acoustic-phonetic corpus. In: Proc. DARPA Speech Recognition Workshop. (1986) 100–109
9. Lee, K.F., Hon, H.: Speaker-independent phone recognition using hidden Markov models. IEEE Trans. Acoust., Speech, Signal Processing **37** (1989) 1641–1648
10. Hirsch, H.G., Pearce, D.: The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In: ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW). (2000) 29–32
11. Kovács, Gy., Tóth, L.: Phone recognition experiments with 2D DCT spectro-temporal features. In: Proc. SACI, IEEE (2011) 143–146
12. Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The HTK Book. Cambridge University Engineering Department, Cambridge (2005)
13. Graves, A., Mohamed, A., Hinton, G.E.: Speech recognition with deep recurrent neural networks. In: Proc. ICASSP. (2013) 6645–6649
14. Tóth, L.: Combining time- and frequency-domain convolution in convolutional neural network-based phone recognition. In: Proc. ICASSP. (2014) 190–194
15. Plahl, C., Sainath, T.N., Ramabhadran, B., Nahamoo, D.: Improved pre-training of deep belief networks using sparse encoding symmetric machines. In: Proc. ICASSP. (2012) 4165–4168
16. Chang, S.Y., Morgan, N.: Robust CNN-based speech recognition with Gabor filter kernels. In: Proc. Interspeech. (2014) 905–909
17. Martinez, A.M.C., Moritz, N., Meyer, B.T.: Should deep neural nets have ears? the role of auditory features in deep learning approaches. In: Proc. Interspeech. (2014) 2435–2439
18. Baby, D., Gemmeke, J.F., Virtanen, T., Van Hamme, H.: Exemplar-based speech enhancement for deep neural network based automatic speech recognition. In: Proc. ICASSP. (2015) 4485–4489

Mély neuronhálós beszédfelismerők GMM-mentes tanítása

Grósz Tamás[1], Gosztolya Gábor[1,2], Tóth László[2]

[1]Szegedi Tudományegyetem, Informatikai Intézet
[2]MTA-SZTE Mesterséges Intelligencia Kutatócsoport
e-mail: { groszt, ggabor, tothl } @ inf.u-szeged.hu

Kivonat Az utóbbi pár évben a beszédfelismerőkben használt rejtett Markov modellekben (hidden Markov model, HMM) az ún. Gauss-keverékmodell (gaussian mixture model, GMM) komponenst leváltották a mély neuronhálók (deep neural network, DNN). Ugyanakkor ezek az új, neuronálókra épülő hibrid HMM/DNN felismerők számos olyan algoritmust megörököltek, melyeket eredetileg GMM-alapú rendszerekhez fejlesztettek ki, és így optimalitásuk az új környezetben nem garantált. A HMM/DNN modellek 'GMM-mentes' tanításához két részfeladatra kell új megoldást adnunk. Az egyik, hogy a mély hálók időben illesztett tanítócímkéket igényelnek, a másik pedig a környezetfüggő állapotok előállítása, amelyre a klasszikus megoldás egy GMM-alapú klaszterezési algoritmus. Bár a HMM/DNN hibridek tanítására léteznek teljes mondatokon dolgozó ún. szekvencia-diszkriminatív tanítóalgoritmusok, ezeket jellemzően csak a tanítás legutolsó fázisában, a modellek finomhangolására szokták bevetni, míg a tanítás elején HMM/GMM modellekel előállított és illesztett címkékből indulnak ki. Jelen cikkünkben viszont megmutatjuk, hogy megfelelő odafigyeléssel a szekvenciatanuló algoritmusok a tanítás legelejétől használhatóak. Az állapotklaszterezési lépésre korábban már javasoltunk egy GMM-mentes megoldást, így a címkeillesztési feladat megoldásával egy teljesen GMM-mentes tanítási sémához jutottunk. Kísérleti eredményeink azt mutatják, hogy a javasolt megoldás nemcsak gyorsabb, mint a hagyományos tanítási módszer, hanem valamivel jobb felismerési pontosságot is eredményez.

Kulcsszavak: mély neurális hálók, szekvencia-diszkriminatív tanítás

1. Bevezetés

A beszédfelismerésben a mély neuronhálók (deep neural network, DNN) áttörésével a hagyományos, Gauss-keverékmodelleken (gaussian mixture model, GMM) alapuló rejtett Markov-modellek (hidden Markov model, HMM) helyett most már az ún. HMM/DNN hibridek számítanak a csúcstechnológiának. Ezen modellek betanítása azonban jelenleg még több ponton is a hagyományos HMM/GMM modellhez kidolgozott tanítási algoritmusokon alapul. Jelenleg a neuronhálós

Grósz Tamást az Emberi Erőforrások Minisztériuma ÚNKP-16-3 kódszámú Új Nemzeti Kiválóság Programja támogatta.

HMM/DNN modell tanítását egy hagyományos HMM/GMM rendszer betanításával kell kezdeni. Ebből a rendszerből nyerjük ki azután azokat a keretszinten illesztett, környezetfüggő állapotcímkéket, amelyek a DNN betanítása során tanítási célként szolgálnak. Ez az eljárás egyrészt erőforrás-pazarló (a HMM/GMM rendszert a tanítócímkék kinyerése után eldobjuk), másrészt semmi sem garantálja, hogy a GMM használatával kialakított és illesztett címkék a DNN számára is optimálisak lesznek. A két feladat – az állapotcímkék időbeli illesztése és környezetfüggő címkékké való konvertálása – közül az utóbbira korábban már adtunk egy GMM-mentes megoldást [1], így ebben a cikkben a másik problémára, azaz az állapotcímkék kezdeti időbeli illesztésére koncentrálunk.

A HMM/DNN modellek DNN komponensének betanítása legegyszerűbben úgy történhet, ha rendelkezésre állnak időben illesztett tanítócímkék, ekkor ugyanis a tanulás során használhatunk olyan klasszikus hibafüggvényeket, mint például a keresztentrópia (cross-entropy, CE). A legtöbbször azonban a tanítóadatokhoz csak mondatszintű átiratokat kapunk, a beszédhangok időbeli illesztése nem áll rendelkezésre. A HMM/GMM modelleknek megvan a tecnhológiája az időbeli illesztések előállítására, melyet gyakran 'flat start' tanításként emlegetnek [2]. Ez az összes beszédhang-modellt azonos paraméterekkel inicializálja, ami lényegében megfelel a hanghatárok időben egyenletes felosztásának. Innen kiindulva a HMM-ek klasszikus Baum-Welch tanítóalgoritmusa iteratívan tanítja és újrailleszti a modell címkéit. Hasonló, iteratív tanításon és újraillesztésen alapuló procedúrát természetesen ki lehet alakítani a DNN-tanításhoz is, akár a jól bevált CE-hibafügvényre építkezve. Senior és tsai. például véletlenszerűen inicializált neuronhálóval teszik ezt [3], míg Zhang és tsai. kiindulásként egyenletes beszédhang-szegmentálást alkalmaznak [4]. Ezek a megoldások működőképesek, de mint látni fogjuk, relatíve lassan konvergálnak, azaz sok tanítási-újraillesztési ciklust igényelnek.

A fenti eljárások megoldják ugyan a címkék illesztését, de továbbra is egy adatkeretek szintjén definiált hibafüggvényt használnak. Ez nem optimális, mivel a felismerés és a kiértékelés is mondatszinten történik. A HMM/GMM-ek körében számos mondatok szintjén definiált, más szóval szekvencia-diszkriminatív hibafüggvényt javasoltak, és ezek jó részét adaptálták is HMM/DNN hibridekre [5,6,7]. A legismertebb ilyen tanítási kritérium a kölcsönös információ maximalizálásán alapuló 'maximum mutual information', vagy röviden MMI-hibafüggvény [5]. A legtöbb szerző azonban a szekvencia-diszkriminatív tanítást csak a tanítási folyamat legvégén, a már betanított modellek finomhangolására alkalmazza. Magyarul, az első lépés mindig egy CE-hibafüggvényen alapuló tanítás (pl. [5,6,8,9,10,11]).

Az ún. 'neuronhálós időbeli osztályozás' (connectionist temporal classification, CTC) az utóbbi néhány évben vált népszerűvé DNN-ek sorozatokon való tanítására olyan esetben, amikor időben illesztett címkék nem állnak rendelkezésre [12]. Rao és tsai. javasoltak is egy 'flat start' tanítási eljárást, amely a CTC-n alapul [13]. A CTC technológiának azonban több hátránya is van az MMI-tanításhoz képest. Először is, a CTC a szokványos állapotcímkék mellett üres címkéket is elhelyez, amelyekkel aztán valamit kezdeni kell később, a környe-

zetfüggő állapotok kialakítása során. Másodszor, a CTC maga nem szekvencia-diszkriminatív módszer, így a legjobb eredményeket akkor adja, ha ilyen hibafüggvényekkel kombinálva használják [12,13].

A korábbi szerzőkkel ellentétben mi egy olyan tanítási eljárásra teszünk javaslatot, amely a tanítás legelejétől kezdve szekvencia-diszkriminatív hibafüggvényt használ. Ehhez a szokványos alkalmazáshoz képest több apró módosításra lesz szükség, amelyeket részletesen bemutatunk. A kísérletek során az általunk javasolt megoldást a Zhang és tsai. cikke alapján megvalósított, CE-hibafüggvényen alapuló iteratív újratanítási-újraillesztési megoldással vetjük össze [4]. Eredményként azt kapjuk, hogy a mi megoldásunk gyorsabb, és az elért szószintű hibaarány is valamivel kisebb. Tanítási módszerünket kombináljuk a korábban javasolt állapotklaszterezési algoritmusunkkal [1], így a végeredményként kapott tanítási eljárás összes lépése mentes lesz a GMM-alapú technológiától.

2. HMM/DNN felismerők 'flat start' tanítása

A HMM/DNN felismerők tanítása előtt egy HMM/GMM rendszert szokás betanítani, és ezzel állíthatóak elő a DNN tanításához szükséges, időben illesztett állapotcímkék. A cikkben két olyan módszert fogunk összehasonlítani, amelyek GMM használata nélkül képesek ugyanezt a feladatot elvégezni. Összehasonlítási alapként egy olyan algoritmus fog szolgálni, amely iteratívan ismétlődő tanítási-újraillesztési ciklusokat végez a HMM/DNN modellel, melynek DNN komponensét hagyományos, keretalapú CE-hibafüggvénnyel tanítja. Saját megoldási javaslatunk ezzel szemben a DNN tanítására szekvencia-diszkriminatív hibafüggvényt fog használni, mégpedig a talán legismertebb ilyet, a korábban már említett MMI-hibafüggvényt [5]. Az MMI-hiba 'flat start' tanításra való használata több apró módosítást fog igényelni, ezeket a 3. fejezetben be fogjuk mutatni.

2.1. Iteratív CE-tanítás és újraillesztés

Az összehasonlítási alapként szolgáló megoldás a CE tanulási kritériumot használja a DNN tanítására oly módon, hogy a címkéket időnként újraillészteni, majd a tanítást megismétli. Az algoritmus implementálása során Zhang és tsai. cikkét próbáltuk követni [4]:

1. A hangfájlokhoz a címkéket egyenletes időközökre bontással rendeljük hozzá, majd betanítjuk a DNN-t.
2. Az aktuális DNN-t használva újraillesztjük a címkéket a HMM/DNN modellel.
3. A régi DNN-t eldobva új hálót tanítunk az új címkehatárokkal.
4. A 2–3 lépéseket konvergenciáig ismételgetjük.

A fenti eljárás végén kapott DNN-t használjuk a címkék időbeli illesztésére, ez alapján a környezetfüggő modellek kialakítására, majd ezek segítségével a végleges DNN betanítására.

A fent ismertetett eljárás előnye, hogy a szokványos CE-hibafüggvény mellett nem igényli új hibafüggvény implementálását a tanításhoz, az újraillesztést pedig standard beszédfelismerési eszközökkel meg lehet oldani. A módszer hátránya, hogy az újratanítás-újraillesztés ismételgetése elég időigényes, amint majd azt a 6. fejezetben látni fogjuk.

2.2. Szekvencia-diszkriminatív tanítás az MMI-hibafüggvénnyel

A hagyományos HMM/GMM modellek szekvencia-diszkriminatív tanítása ma már sztenderdnek számít. Többféle hibafüggvényt is javasoltak e célra [14], és ezeket már a HMM/DNN modellekre is átültették [5,6,10,15]. A legrégebbi és legegyszerűbb ilyen hibakritérium a maximális kölcsönös információ (maximum mutual information, MMI) hibafüggvény. Az MMI függvény a jellemzővektor-sorozat és a hozzárendelt allapotsorozat kölcsönös információját méri. A jellemzővektorok sorozatára az $O_u = o_{u1}, \ldots, o_{uT_u}$, az u mondathoz tartozó szósorozatra pedig a W_u jelölést használva, az MMI-hibafüggvényt az alábbi módon formalizálhatjuk:

$$F_{MMI} = \sum_u \log \frac{p(O_u|S_u)^\alpha p(W_u)}{\sum_W p(O_u|S)^\alpha p(W)}, \tag{1}$$

ahol $S_u = s_{u1}, \ldots, s_{uT_u}$ a W_u-hoz tartozó állapotsorozat, α pedig az akusztikus modell súlya. A nevezőben található összegzés az u mondatra felismerési kimenetként kapott legvalószínűbb beszédhang-sorozatokat tartalmazza – ezt úgy kaphatjuk meg, hogy egyetlen kimenet helyett ún. szóhálót (lattice) generáltatunk a felismerővel. Az (1) egyenletet deriválva a $\log p(o_{ut}|r)$ log-likelihood érték szerint r állapotban és t időpillanatban, azt kapjuk, hogy

$$\begin{aligned} \frac{\partial F_{MMI}}{\partial \log p(o_{ut}|r)} &= \alpha\delta_{r;s_{ut}} - \frac{\alpha \sum_{W:s_t=r} p(O_u|S)^\alpha p(W)}{\sum_W p(O_u|S)^\alpha p(W)} \\ &= \alpha\big(\delta_{r;s_{ut}} - \gamma_{ut}^{DEN}(r)\big), \end{aligned} \tag{2}$$

ahol $\gamma_{ut}^{DEN}(r)$ a t időpillanatban az r állapotban való tartózkodás valószínűsége a nevezőhöz tartozó felismerési szóhálón számolva – amit a HMM-ek szokványos 'előre-hátra' algoritmusával kaphatjuk meg –, a $\delta_{r;s_{ut}}$ pedig a Kronecker-delta függvény (ez adja meg a 0-1 jellegű tanítási célvektorokat).

3. Flat start tanítás az MMI-hibakritériummal

A szekvencia-diszkriminatív tanítási kritériumokat, így például az MMI hibafüggvényt mostanra már széles körben használják a HMM/DNN hibridek tanítására. Tapasztalatunk szerint azonban a tanítást minden szerző a CE-hibakritériummal kezdi el, és a szekvencia-diszkriminatív hibakritériumot csak a tanítás végső fázisában vetik be, pusztán a modellek finomhangolására használva

azt [6,10]. Ez esetben viszont a CE-tanítás miatt mindenképpen szükség van valamilyen módszerre az időillesztett tanítási célvektorok előállítására. Ezekkel a szerzőkkel szemben mi azt állítjuk, hogy az MMI célfüggvényt rögtön a tanítás elejétől kezdve lehet használni, így a CE-tanulás, illetve ezáltal az ehhez szükséges illesztett címkék előállítása kihagyható. A módszerünk működőképessége érdekében az alábbi apró változtatásokat kellett elvégeznünk.

Elsőként, a (2) egyenlet számlálójában a $\delta_{r;s_{ut}}$ értékek helyett a $\gamma_{ut}^{NUM}(r)$ értékeket fogjuk használni, amit az előre-hátra algoritmussal számolunk ki. Ennek előnye, hogy bináris értékek helyett 0-1 közötti valószínűségi értékekkel dolgozhatunk, így kihagyhatjuk a (szokásosan GMM-alapú) címkeillesztési lépést. Ezt a megoldási lehetőséget több tanulmányban is említik (pl. [6,15]), de egyedül Zhou és tsai. cikkében találtuk nyomát, hogy valaki meg is valósította [8]. Azonban a tanítási folyamatot ők is CE-tanítással indítják, azaz az általunk javasolt flat start MMI-tanítást nem próbálják ki.

Mivel a szekvencia-diszkriminatív tanítási kritériumot a kész rendszer finomítására szokták használni, az MMI-célfüggvényt a teljes felismerővel, azaz környezetfüggő beszédhang-modellek és szószintű nyelvi modell mellett számolják ki. A (2) egyenlet nevezőjének kiszámolása a teljes felismerési procedúra lefuttatását igényli, ami a teljes modell használata mellett nagyon lassú. Emiatt a számlálóhoz és nevezőhöz szükséges hálók leszámolását csak egyszer szokták elvégezni, méghozzá az MMI-tanítás elindítása előtt. Ezzel szemben mi a szekvencia-diszkriminatív tanulást szószintű helyett pusztán fonetikai szintű szótárral végezzük, ráadásul környezetfüggő helyett környezetfüggetlen beszédhang-modellekkel. E két változtatás nagyon gyors dekódolást tesz lehetővé, így a számlálót és nevezőt minden egyes mondat után újra tudjuk számolni. Ez a módosítás kulcsfontosságú az eljárásunk gyors konvergenciája szempontjából. A szószintű átiratok fonetikai átirattá konvertálására a HTK rendszerben javasolt technikát használtuk, azaz első körben a hangsorozatot az egyes szavak fonetikai átiratát behelyettesítve kapjuk meg, a szavak közé sehol sem rakunk csendet. Az esetleges kiejtésvariánsokat, illetve a szavak közti csendet néhány iteráció után illesztjük be, újraillesztést végezve a már relatíve elfogadható szinten betanult modellel [2].

További finomítás, hogy a fonetikai dekódolás során nem használjuk sem a hangok a priori valószínűségét, sem bigramot vagy egyéb, összetettebb nyelvi modellt, emiatt a (2) egyenletből az α tag is elhagyható. Emellett, a számítási igény további csökkentése érdekében a $\gamma_{ut}^{DEN}(r)$ érték közelítésére a hálózat összes útvonalának figyelembe vétele helyett csak a legvalószínűbb felismerési útvonalat használtuk fel (ezt a közelítést jelöli a $\hat{\gamma}_{ut}^{DEN}(r)$ formula).

Ezekkel a módosításokkal a célfüggvény gradiense az alábbi módon alakul:

$$\begin{aligned}\frac{\partial F_{MMI}}{\partial a_{ut}(s)} &= \sum_r \frac{\partial F_{MMI}}{\partial \log p(o_{ut}|r)} \frac{\partial \log p(o_{ut}|r)}{\partial a_{ut}(s)} \\ &= \gamma_{ut}^{NUM}(s) - \hat{\gamma}_{ut}^{DEN}(s),\end{aligned} \tag{3}$$

amit pedig már közvetlenül tudunk használni a DNN tanítása során. Neuronhálók tanításánál jól ismert technika, hogy a tanítóhalmaz egy kis részét félretesszük validálási célra. Ha az aktuális tanítási iteráció után a hiba növekedne,

(1) A keretek tanítási célértékét ($\gamma_{ut}^{NUM}(r)$-t) az előre-hátra algoritmussal határozzuk meg.
(2) Beszédhang-szintű átiratokkal és környezetfüggetlen beszédhang-modellekkel dolgozunk.
(3) Nem használunk a priori valószínűségeket, sem nyelvi modellt.
(4) $\gamma_{ut}^{DEN}(r)$ értékét a legvalószínűbb felismerési útvonal valószínűségével ($\hat{\gamma}_{ut}^{DEN}(r)$) közelítjük.
(5) A tanítás hibáját a validációs halmazon mérjük, és ha ez a hiba növekedne, akkor visszatérünk az iteráció előtti paraméterekhez, viszont csökkentjük a tanulási rátát.

1. táblázat. *A 'flat start MMI' tanításhoz javasolt módosításaink összegzése.*

akkor a súlyokat viszaállítjuk az iteráció előttire, és a tanítást innen folytatjuk egy kisebb tanulási rátával. Ez a módszer szekvencia-diszkriminatív tanítás esetén is természetes módon alkalmazható [5], sőt, úgy találtuk, hogy a flat-start tanítási módszerünk stabilitásában ennek a lépésnek nagyon fontos szerepe van, mivel segít elkerülni az elakadásokat.

Az MMI-kritérium használatához javasolt módosításainkat a 1. táblázatban összszegezzük. Az (1)-(4) módosítási javaslatok egyrészt gyorsítják a felismerési folyamatot, másrészt növelik annak hibákkal szembeni robusztusságát. A (2) pont kulcsfontosságú szerepű abban, hogy a szekvencia-diszkriminatív tanulást a tanulási folyamat elejétől, még a környezetfüggő modellek kialakítása előtt alkalmazni tudjuk. Végezetül, az (5) pont segít az elakadási problémák kikerülésében, feloldásában.

4. KL-divergencia alapú állapotkapcsolás

Amikor a flat start tanítás konvergált, azaz megkaptuk a környezetfüggetlen (context-independent, CI) modellek legjobb időbeli illesztését, következhet a környezetfüggő (context-dependent, CD) modellek kialakítása. Jelenleg erre a legelterjedtebb megoldás az ún. döntési fa-alapú állapotklaszterező algoritmus [16]. Ez az algoritmus összegyűjti az egyes beszédhang-állapotok összes, különböző kontextusokban előforduló példányát, majd minden egyes csomópontban kettéosztva ezt a halmazt, felépít egy döntési fát, bizonyos előredefiniált kérdéseket követve. A kettéosztáshoz Gauss-görbét illeszt az aktuális adatok eloszlására, majd az alapján a kérdés alapján osztja ketté a csomópontot, amelyik a legnagyobb növekedést eredményezi a Gauss-görbék illeszkedésében (likelihood-értékében). Habár ez az algoritmus remekül működik GMM-alapú akusztikus modellek esetén, megkérdőjelezhető, hogy a Gauss-görbék illeszkedése mennyire alkalmas a mély neuronhálókkal való megtanulhatóság mérésére.

A fentiek miatt javasoltunk egy olyan alternatív megoldást, amely Gauss-görbék illesztése helyett betaníthat egy segéd-neuronhálót, majd ennek kimeneti

értékei alapján végzi el a döntési fa felépítését. Mivel a neuronháló-kimenetek egy diszkrét valószínűségi eloszlásból vett mintáknak tekinthetők, ezen kimeneti vektorok összehasonlítására természetes módon adódik az ún. Kullback-Leibler (KL) divergencia. Így az állapotklaszterezési algoritmust vezérlő, Gauss-görbékre felírt távolságfüggvényt lecseréltük egy KL-divergencián alapuló döntési kritériumra, Imseng és társainak cikkét követve [17]. A döntési függvény lecserélésén túl a döntésifa-építési mechanizmus változatlan marad, így a korábbi implementációk könnyen módosíthatók. Ezzel a megoldással nemcsak eliminálták a Gauss-görbéket az állapotklaszterezési folyamatból, de még 4% relatív javulást is elértünk a szószintű hibában. Az algoritmus részleteit korábban már publikáltuk, lásd [1].

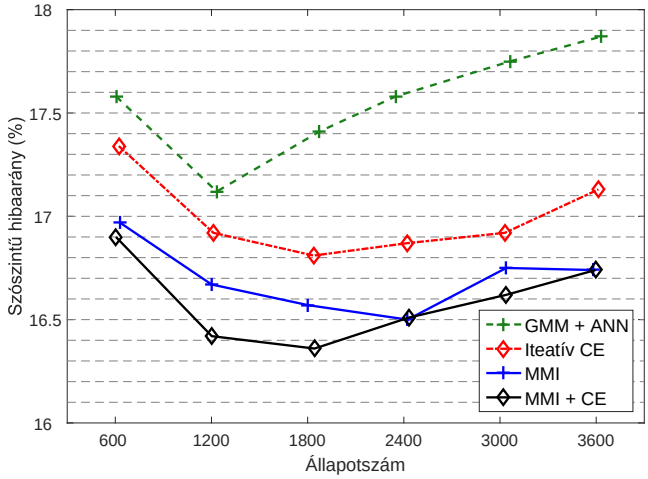
5. Kísérleti beállítások

Kísérleteink paraméterezése lényegében megegyezik a korábbi cikkeinkben leírtakkal [1]. Akusztikus modellként egy öt rejtett rétegű mély neuronhálót használtunk, melynek minden rétege 1000 'egyenirányított' (rectifier) neuront tartalmazott [18], míg a kimeneti rétegben softmax aktivációs függvényt alkalmaztunk. A modell saját neuronhálós csomagunkra épült, mellyel korábban kiemelkedő eredményeket értünk el több különböző feladaton is ([19,20,21,22]). Jellemzőkészletként egy 40-sávos mel-szűrőkészlet energiakimeneteit használtuk, a szokványos első és második derivált értékeivel kiegészítve. A felismerést és kiértékelést a HTK programcsomag mély hálókhoz igazított verziójával végeztük [2].

Beszédkorpuszként a 'Szeged' híradós beszédadatbázist használtuk, amely 28 órányi híradófelvételt tartalmaz nyolc tévécsatornáról rögzítve [23]. Tanítóhalmazként egy kb. 22 órányi részt különítettünk el, míg 2 órányi adatot használtunk validációs avagy fejlesztési (development) halmazként, 4 órányit pedig tesztelésre. Nyelvi modellként egy sztenderd trigram modell szolgált, a kiejtési szótár szűk ötszázezer szóalakot tartalmazott. Az állapotklaszterző algoritmus paramétereit úgy állítottuk be, hogy a különböző kísérletekben nagyjából 600, 1200, 1800, 2400, 3000, illetve 3600 kapcsolt állapotot kapjunk.

A beszédhang-modellek kezdeti illesztésére négyféle módszert próbáltunk ki és hasonlítottunk össze. Elsőként egy hagyományos, GMM-alapú rendszert tanítottunk be, és ezzel állítottuk elő az időben illesztett CI címkéket. Ezután az így kapott állapotcímkéken betanítottunk egy szimpla (azaz nem mély) neuronhálót a CE-kritériummal, és az így kapott hálóval újraillesztettük a címkéket (korábbi tanulmányunkban azt kaptuk, hogy szimpla helyett mély hálót használva nem javulnak az eredmények [1]). A táblázatokban erre a módszerre *„GMM + ANN"* jelöléssel fogunk hivatkozni. Az újraillesztés után a CD modellek előállítására mind a GMM-alapú, mind a KL-kritérium alapú megoldást kipróbáltuk, ahol az utóbbi esetben természetesen a neuronháló kimenete szolgált inputként.

Míg a fenti megoldás egy GMM-alapú rendszerből indult ki, 'GMM-mentes' megoldásként a 2. és 3. fejezetekben ismertetett algoritmusokat vetettük be. Ezekben a kísérletekben a neuronháló mindig mély háló volt, öt rejtett réteggel. Az iteratív CE-tanításon és újraillesztésen alapuló módszer esetében (a táblázat-



1. ábra. Szószintű hibaarány a KL-klaszterezéssel kapott állapotok számának függvényében, a fejlesztési halmazon.

ban *„Iteratív CE”*) négy tanítási-újraillesztési ciklust futtattunk, az ezt követő állapotklaszterezés során pedig a KL-divergencia alapú módszert hajtottuk végre a végső neuronháló által adott illesztésen. Az MMI-tanítás esetén (a táblázatban *„MMI”*) szintén véletlen súlyokkal inicializált mély hálóból indultunk ki, melyet a korábban ismertetett módon tanítottunk. A végeredményként előálló DNN szolgáltatta az inputot a rákövetkező, KL-divergencia alapú klaszterezési lépéshez. Végezetül, a negyedik kísérletben a szekvencia-diszkriminatív MMI-tanítással kapott illesztett címkéken lefuttattunk még egy CE-tanítást, és ennek kimenetén végeztük el a KL-kritérium alapú klaszterezést (*„MMI + CE”*). Tettük ezt azért, mert azt tapasztaltuk, hogy a CE, illetve az MMI kritérium eléggé eltérő valószínűségi eloszlásokat eredményez, ezért kíváncsiak voltunk, hogy vajon a klaszterezést ez hogyan befolyásolja.

Cikkünk fő célja a ‘flat-start’ lépés, azaz a kezdeti címkeillesztéseket előállító lépés különböző változatainak összehasonlítása volt. Ezért az állapotklaszterezés után előálló CD-modelleket már csak az egyszerűbb CE-kritériummal tanítottuk. Természetesen ezeket a modelleket tanítás után tovább lehetne finomítani a szekvencia-diszkriminatív tanítás bevetésével. Ezzel vélhetően kicsit jobb eredményeket kapnánk ugyan, de mivel ez egy sztenderd eljárás, ezért ettől jelen cikkben eltekintettünk.

6. Kísérleti eredmények

A különböző módszerekkel kapott szószintű hibaarányok alakulását a fejlesztési halmazon az 1. ábra mutatja, különböző állapotszámok esetére. Mint látható, a GMM-alapú módszer messze a legrosszabbul teljesített, míg az MMI-alapú

Flat start módszer	Állapotkapcsolási módszer	Szóhiba (%)		Iterációk száma
		Dev.	Teszt	
GMM + ANN	GMM	18.83%	17.27%	—
GMM + ANN	KL	17.12%	16.54%	—
Iteratív CE	KL	16.81%	16.50%	48
MMI		16.50%	15.96%	13
MMI + CE		16.36%	15.86%	29

2. táblázat. *Szószintű hibaarány a különféle 'flat start' illetve állapotkapcsolási stratégiák esetén.*

flat start eljárás minden esetben kissé jobb eredményeket adott, mint az iteratív megoldás. Habár az MMI-t követő CE tanítás (az 'MMI+CE'-vel jelölt modell) kisebb állapotszám mellett némileg jobb eredményeket adott, ez a javulás nem jelentős annyira, hogy megérje a többletidőt. Mindez azt mutatja, hogy a szekvencia-diszkriminatív tanítás egyaránt pontos időillesztéseket és jó valószínűségi becsléseket eredményez.

A 2. táblázat összesíti a különböző konfigurációkkal elért legjobb szóhibaarányokat a fejlesztési és teszthalmazokon. Az állapotklaszterezési módszerek közül a KL-divergencia alapú megoldás minden esetben egyértelműen túlszárnyalta a GMM-alapú módszert. Az illesztési technikákat összevetve azt láthatjuk, hogy a HMM/GMM rendszerre támaszkodó megoldás bizonyult a legrosszabbnak, amin a neuronhálós újraillesztés sem segített. Az iteratív CE-alapú tanítási módszer kicsivel rosszabb lett a két MMI alapú megoldásnál. E módszer esetén sajnos elég nehéz megmondani az optimális iterációszámot. Zhang és társai 20 lépésen át végezték az iterációt [4], míg mi csak 4 lépésig futtattuk. Emiatt érdemes a futási időket is összevetni, mely értékek a 2. táblázat jobb szélső oszlopában láthatók (a tanítási iterációk számát a „GMM + ANN" rendszer esetében nem tüntettük fel, mivel ott a tanítás egy radikálisan eltérő procedúrán alapult). Az iteratív CE-tanítás 4 iterációt igényelt, összesen 48 DNN-tanítási ciklust eredményezve, míg az MMI-tanítás ennek csak kb. a negyedét. Habár az utóbbihoz az előre-hátra algoritmus lefuttatásának költségét is hozzá kell adni, ezzel együtt is egyértelmű, hogy az MMI-tanítás műveletigénye jóval kisebb.

Ha a futási időt DNN-tanítási ciklusok helyett egyszerűen CPU/GPU időben mérjük, akkor még nagyobb különbségeket kapunk az MMI módszer javára (3 óra 16-tal szemben). Ennek oka, hogy a CE-tanítás során 100-as minibatch-méretet használtunk, míg az MMI-tanítás során a kötegméret az egyes felvételek méretével egyezett meg, ami átlagosan 1000 körüli batch-méretet, és így a GPU-k struktúrája miatt gyorsabb végrehajtást eredményezett.

Álláspontunk szerint módosításaink közül kettő kulcsfontosságú a javasolt algoritmusunk sebessége és futásideje szempontjából. Az első módosítás, hogy az illesztést környezetfüggetlen beszédhang-modellekkel, nyelvi modell nélkül végezzük. Ez teszi lehetővé a gyors számítást, és így a célfüggvényben található

szóhálók frissítését minden egyes mondat feldolgozása után. Az irodalomban egyetlen olyat cikket találtunk, amely nem csak a tanulási iterációk végén frissíti ezeket a hálókat, ebben a cikkben azonban egy masszívan párhuzamosított architektúrát írnak le, ami nagyon nehezen összevethető a mi szekvenciális algoritmusunkkal [24].

A stabilitást illetően közismert, hogy a szekvencia-diszkriminatív módszerek erősen hajlamosak a túltanulásra. Az állapotcímkék és azok illesztésének egyidejű tanulása gyakran vezet az ún. „run-away silence model" esetéhez, amikor a hosszú csendszakaszok miatt a csendhez tartozó kimenet egyre dominánsabbá válik, majd az illesztést is elrontva 'megeszi' a beszédhang-szakaszokat is [25]. A hasonló esetek elkerülésére egy független validációs halmazon mértük a neuronháló hibáját, és ha a hiba az aktuális iteráció után megugrott, akkor a korábbi súlyok visszaállítása után egy kisebb tanulási rátával újrapróbáltuk a tanulást. Tapasztalatunk szerint ez az egyszerű trükk sokat segített a hasonló elakadási jelenségek megakadályozásában.

7. Konklúzió

Cikkünkben megmutattuk, hogy a HMM/DNN modellek szekvencia-diszkriminatív tanítását a tanítás legelső, ún. 'flat start' fázisában is sikeresen lehet használni. E célra a szokványos MMI tanítási kritériumot alkalmaztuk, míg a tanítási folyamatban néhány apró módosítást vezettünk be. Kísérleti eredményeink azt mutatták, hogy – a CE tanítási kritériumon alapuló újratanítás-újraillesztés stratégiával összevetve – az általunk javasolt megoldás lényegesen gyorsabb, és még a szóhiba-arányt is csökkenti valamelyest. A korábban javasolt KL-divergencia alapú állapotklaszterezési megoldást is bevonva, összességében egy olyan HMM/DNN tanítási algoritmust adtunk, amely egyáltalán nem igényli a hagyományos HMM/GMM modellek használatát.

Hivatkozások

1. Gosztolya, G., Grósz, T., Tóth, L., Imseng, D.: Building context-dependent DNN acousitc models using Kullback-Leibler divergence-based state tying. In: Proceedings of ICASSP. (2015) 4570–4574
2. Young, S., Evermann, G., Gales, M.J.F., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The HTK Book. Cambridge University Engineering Department, Cambridge, UK (2006)
3. Senior, A., Heigold, G., Bacchiani, M., Liao, H.: GMM-free DNN acoustic model training. In: Proceedings of ICASSP. (2014) 5639–5643
4. Zhang, C., Woodland, P.: Standalone training of context-dependent Deep Neural Network acoustic models. In: Proceedings of ICASSP. (2014) 5634–5638
5. Kingsbury, B.: Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling. In: Proceedings of ICASSP. (2009) 3761–3764
6. Veselý, K., Ghoshal, A., Burget, L., Povey, D.: Sequence-discriminative training of deep neural networks. In: Proceedings of Interspeech. (2013) 2345–2349

7. Grósz, T., Gosztolya, G., Tóth, L.: A sequence training method for Deep Rectifier Neural Networks in speech recognition. In: Proceedings of SPECOM, Novi Sad, Serbia (2014) 81–88
8. Zhou, P., Dai, L., Jiang, H.: Sequence training of multiple Deep Neural Networks for better performance and faster training speed. In: Proceedings of ICASSP. (2014) 5664–5668
9. Saon, G., Soltau, H.: A comparison of two optimization techniques for sequence discriminative training of Deep Neural Networks. In: Proceedings of ICASSP. (2014) 5604–5608
10. Wiesler, S., Golik, P., Schüter, R., Ney, H.: Investigations on sequence training of neural networks. In: Proceedings of ICASSP. (2015) 4565–4569
11. Chen, D., Mak, B., Sivadas, S.: Joint sequence training of phone and grapheme acoustic model based on multi-task learning Deep Neural Networks. In: Proceedings of Interspeech. (2014) 1083–1087
12. Graves, A., Mohamed, A.R., Hinton, G.E.: Speech recognition with Deep Recurrent Neural Networks. In: Proceedings of ICASSP. (2013) 6645–6649
13. Rao, K., Senior, A., Sak, H.: Flat start training of CD-CTC-SMBR LSTM RNN acoustic models. In: Proceedings of ICASSP, Shanghai, China (2016) 5405–5409
14. He, X., Deng, L.: Discriminative Learning for Speech Recognition. Morgan & Claypool, San Rafael, CA, USA (2008)
15. Yu, D., Deng, L.: Chapter 8: Deep neural network sequence-discriminative training. In: Automatic Speech Recognition — A Deep Learning Approach. Springer (2014)
16. Young, S.J., Odell, J.J., Woodland, P.C.: Tree-based state tying for high accuracy acoustic modelling. In: Proceedings of HLT. (1994) 307–312
17. Imseng, D., Dines, J.: Decision tree clustering for KL-HMM. Technical Report Idiap-Com-01-2012, IDIAP Research Institute (2012)
18. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier networks. In: Proceedings of AISTATS. (2011) 315–323
19. Tóth, L.: Convolutional deep maxout networks for phone recognition. In: Proceedings of Interspeech. (2014) 1078–1082
20. Grósz, T., Busa-Fekete, R., Gosztolya, G., Tóth, L.: Assessing the degree of nativeness and Parkinson's condition using Gaussian Processes and Deep Rectifier Neural Networks. In: Proceedings of Interspeech. (2015) 1339–1343
21. Tóth, L., Gosztolya, G., Vincze, V., Hoffmann, I., Szatlóczki, G., Biró, E., Zsura, F., Pákáski, M., Kálmán, J.: Automatic detection of mild cognitive impairment from spontaneous speech using ASR. In: Proceedings of Interspeech, Dresden, Germany (2015) 2694–2698
22. Kovács, Gy., Tóth, L.: Joint optimization of spectro-temporal features and Deep Neural Nets for robust automatic speech recognition. Acta Cybernetica **22** (2015) 117–134
23. Grósz, T., Tóth, L.: A comparison of Deep Neural Network training methods for Large Vocabulary Speech Recognition. In: Proceedings of TSD, Pilsen, Czech Republic (2013) 36–43
24. Bacchiani, M., Senior, A., Heigold, G.: Asynchronous, online, GMM-free training of a context dependent acoustic model for speech recognition. In: Proceedings of Interspeech, Singapore, Singapore (2014) 1900–1904
25. Su, H., Li, G., Yu, D., Seide, F.: Error back propagation for sequence training of context-dependent deep networks for conversational speech transcription. In: Proceedings of ICASSP. (2013) 6664–6668

Beszédszintézis ultrahangos artikulációs felvételekből mély neuronhálók segítségével

Csapó Tamás Gábor[1,2], Grósz Tamás[3], Tóth László[4], Markó Alexandra[2,5]

[1]Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék,
[2]MTA-ELTE Lendület Lingvális Artikuláció Kutatócsoport,
[3]Szegedi Tudományegyetem, Informatikai Intézet,
[4]MTA-SZTE Mesterséges Intelligencia Kutatócsoport,
[5]Eötvös Loránd Tudományegyetem, Fonetikai Tanszék,

e-mail: csapot@tmit.bme.hu, groszt@inf.u-szeged.hu,
tothl@inf.u-szeged.hu, marko.alexandra@btk.elte.hu

Kivonat A kutatás célja egy olyan rendszer létrehozása, amely a nyelv ultrahangos felvételeiből beszédet tud szintetizálni. A kutatás során egy női beszélőtől rögzítettünk közel 200 bemondáshoz tartozó szinkronizált akusztikai és artikulációs adatot, azaz nyelvultrahang-felvételt. A beszédből az alapfrekvenciát és spektrális paramétereket nyertük ki. Ezután mély neurális hálón alapuló gépi tanulást alkalmaztunk, melynek bemenete a nyers nyelvultrahang volt, kimenete pedig a beszéd spektrális paraméterei, ún. ,,mel-általánosított kepsztrum" reprezentációban. A tesztelés során egy impulzus-zaj gerjesztésű vokódert alkalmaztunk, mellyel az eredeti beszédből származó F0 paraméterrel és a gépi tanulás által becsült spektrális paraméterekkel mondatokat szintetizáltunk. Az így szintetizált beszédben sok esetben szavak, vagy akár teljes mondatok is érthetőek lettek, így a kezdeti eredményeket biztatónak tartjuk.

Kulcsszavak: gépi tanulás, artikuláció, beszédtechnológia, vokóder

1. Bevezetés

A beszédhangok az artikulációs szervek (hangszalagok, nyelv, ajkak stb.) koordinált mozgásának eredményéből állnak elő. Az artikuláció és a keletkező beszédjel kapcsolata régóta foglalkoztatja a beszédkutatókat. Beszéd közben a nyelv mozgását többféle technológia segítségével is lehet rögzíteni és vizsgálni, például röntgen [1,2,3], ultrahang [4,5], elektromágneses artikulográf (EMA) [6,7], mágnesesrezonancia-képalkotás (MRI) [8,9] és permanens mágneses artikulográf (PMA) [10]. Az ultrahangos technológia előnye, hogy egyszerűen használható, elérhető árú, valamint nagy felbontású (akár 800 x 600 pixel) és nagy sebességű (akár 100 képkocka/s) felvétel készíthető vele. A hátránya viszont az, hogy a hagyományos beszédkutatási kísérletekhez a rögzített képsorozatból ki kell nyerni a nyelv és a többi beszédszerv körvonalát ahhoz, hogy az adatokon további vizsgálatokat lehessen végezni. Ez elvégezhető manuálisan, ami

rendkívül időigényes, illetve automatikus módszerekkel, amelyek viszont ma még nem elég megbízhatóak [11]. Arra is lehetőség van, hogy az ultrahangképekből közvetlenül, a nyelvkontúr kinyerése nélkül állapítsunk meg az artikulációs szerv aktuális pozíciójára utaló információt [12].

Az artikuláció és az akusztikai kimenet kapcsolatát gépi tanulás alapú eszközökkel is vizsgálták már. Az artikuláció-akusztikum konverzió eredményei a szakirodalomban elsősorban az ún. 'Silent Speech Interface' (SSI, magyarul 'némabeszéd-interfész') rendszerek fejlesztéséhez járulnak hozzá [13]. Az SSI lényege, hogy az artikulációs szervek hangtalan mozgását felvéve a gépi rendszer ebből beszédet szintetizál, miközben az eszköz használója valójában nem ad ki hangot. Ez egyrészt a beszédsérült embereknek (pl. gégeeltávolítás után) lehet hasznos, másrészt potenciálisan alkalmazható zajos környezetben történő beszédhang kiadására, kiabálás nélkül. Mivel az SSI közvetlenül az artikulációt rögzíti, ezért a rendszer nem érzékeny a környezeti zajokra. A konverziós feladathoz többnyire EMA-t [14,15,16], ultrahangot [17,18,19,20,21,22] vagy PMA-t [23] használnak inputként, mi azonban csak az ultrahangra koncentrálunk a jelen áttekintésben.

Az egyik első hasonló kísérletben egy egyszerű neurális hálózattal próbálták a nyelvmozgás ultrahangos képének és a beszéd spektrális paramétereinek összefüggését megtalálni [17], de az eredmények ekkor még nem voltak meggyőzőek, mert az alkalmazott neurális hálózat nem volt alkalmas a komplex feladat megoldására. Később az SSI rendszereket „felismerés-majd-szintézis" alapon valósították meg, azaz a cél az volt, hogy az ultrahangalapú artikulációs adatokból először a beszédhangokat kinyerjék egy vizuális felismerő módszerrel, majd ezután egy beszédszintézis-rendszer felolvassa a beszédet [18]. Ezen megoldás hátránya, hogy a komponensek hibája összeadódik, azaz a beszédhang-felismerés esetleges tévesztése nagyon elrontja a beszédszintézis eredményét. A későbbi SSI rendszerekben ezért a „közvetlen szintézis" módszer terjedt el, azaz a köztes beszédhangfelismerés nélkül, az artikulációs adatok alapján próbálják megbecsülni a beszéd valamilyen reprezentációját (tipikusan a spektrális paramétereit) [19,20,21]. Az alkalmazott gépi tanulási módszer ezekben a kísérletekben Gauss-keverékmodell (gaussian mixture model, GMM) [19], illetve rejtett Markov-modell volt [20,21].

A legújabb eredmények szerint a mély neurális hálózatok (például a konvolúciós hálózatok) az emberi teljesítményt megközelítő vagy akár jobb pontosságot értek el olyan feladatokban, mint az objektumfelismerés [24], képek osztályozása [25], él/kontúr-detekció [26] stb. Az ultrahangalapú SSI témakörében eddig egyetlen kutatás alkalmazott mély neurális hálózatot [22]. A kutatásban ultrahang- és ajakvideó-alapú artikulációs adatok alapján alkalmaztak autoencoder neuronhálózatot, illetve előrecsatolt hálózatot (MLP) egy egyszerű vokóder spektrális (egész pontosan ún. LSF) paramétereinek becslésére, végül ez alapján éneket hoztak létre egy artikulációs szintetizátorral. Az eredmények és a hangminták szerint a becslési feladat megoldása előremutató, de még további kutatást igényel.

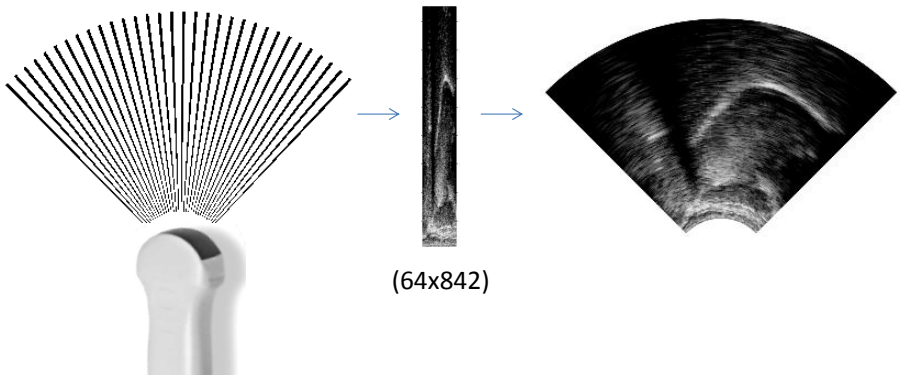
1.1. A jelen kutatás célja

A szakirodalmi áttekintés szerint az artikuláció-akusztikum konverzió még kezdeti stádiumban van, és a valós időben működő SSI rendszerek kifejlesztése a feladat minél pontosabb megoldását igényli. A jelen tanulmányban bemutatjuk az első erre irányuló kísérletünket, amelyben egy magyar beszélő ultrahangos felvételei alapján beszédet szintetizálunk.

2. Módszerek

2.1. Felvételek és adatok

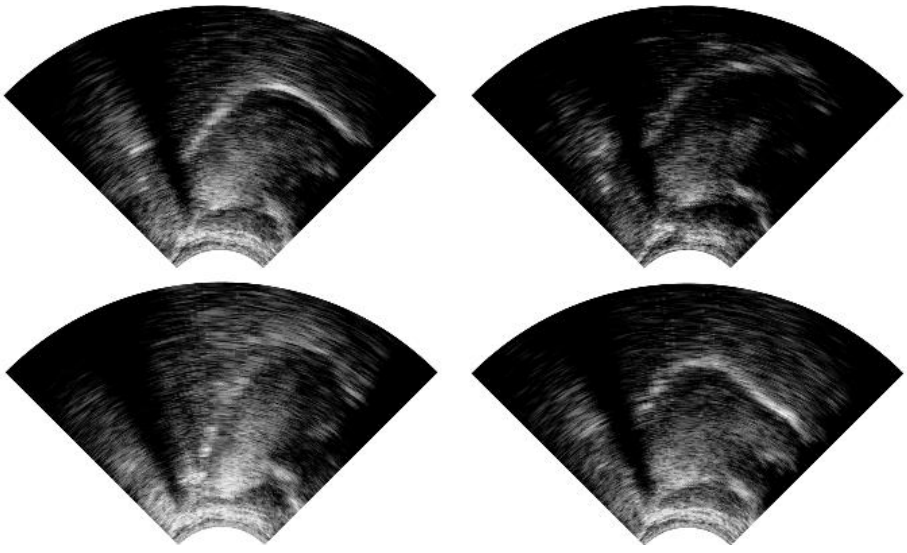
A kutatáshoz egy női beszélőtől (MA) rögzítettünk párhuzamos ultrahang- és beszédfelvételeket. A felvételek az ELTE Fonetikai Tanszék egyik csendes szobájában készültek, a szakirodalomban javasolt helyzetben és beállításokkal [5]. A beszélő a PPBA adatbázis [27] első 176 mondatát olvasta fel. A nyelv középvonalának (szagittális) mozgását a SonoSpeech rendszerrel rögzítettük (Articulate Instruments Ltd.) egy 2–4 MHz frekvenciájú, 64 elemű, 20 mm sugarú konvex ultrahang-vizsgálófejjel, 82 fps sebességgel. A felvételek során ultrahang-rögzítő sisakot is használtunk (Articulate Instruments Ltd., fénykép: [28]). A beszédet egy Audio-Technica – ATR 3350 omnidirekcionális kondenzátor-mikrofonnal rögzítettük, amely a sisakra volt csípteve, a szájtól kb. 20 cm-re. A hangot 22050 Hz mintavételi frekvenciával digitalizáltuk egy M-Audio – MT-RACK PLUS hangkártyával. Az ultrahang és a beszéd szinkronizációja a SonoSpech rendszer 'Frame sync' kimenetét használva történt: minden elkészült ultrahangkép után ezen a kimeneten megjelenik egy néhány ns nagyságrendű impulzus, amelyet egy 'Pulse stretch' egység szélesebb négyszög ugrássá alakít, hogy digitalizálható legyen [28]. Ez utóbbi jelet szintén a hangkártya rögzítette. A felolvasandó mondatok képernyőn megjelenítését és az adatok felvételét a kísérlet vezetője végezte az Articulate Assistant Advanced (Articulate Instruments Ltd.)



1. ábra. Nyers adatokból ultrahangkép előállítása.

szoftver használatával. A ultrahangból származó nyers adatokat ezután közvetlenül bináris formátumba mentettük (így nem veszett el adat a képpé konvertálás során). Az 1. ábra mutatja, hogy a letapogatás hogyan történik a SonoSpeech rendszerrel: az ultrahangfej 64 radiális vonalon (bal oldalon), minden vonalon 842 helyen méri az intenzitást, és a nyers adatban minden intenzitásértéket 8 biten tárol (ennek eredménye látható középen). Ha ezt a szokásos ultrahangképpé akarjuk alakítani, akkor az adatokat poláris koordinátarendszerben lehet ábrázolni szürkeárnyalatos képként, mely a jobb oldalon látható.

A 2. ábra néhány példát mutat a nyelvről készített ultrahangfelvételre a fenti női beszélőtől. A felvételeken bal oldalon látható a nyelvgyök, jobb oldalon a nyelvhegy; a kettő között a nyelv felső felülete. A bal oldali sötétebb rész a nyelvcsont helyére, míg a jobb oldali sötétebb rész az állkapocscsont helyére utal (mivel az ultrahang-hullám a csontokon nem tud áthatolni). A felvételek során az ultrahang-vizsgálófejet az áll alá helyeztük; így az ultrahangjelben a legnagyobb változást a nyelv izomzatának felső határa okozza, ami az ultrahangos képeken ideális esetben jól kivehető fehér sávot eredményez. Mivel a hullámok nagy része nem jut tovább a nyelv felső határán, így a távolabbi szövetpontokról, a szájpadlásról kevesebb az információnk. A 2. ábrán az is látható, hogy a képek minősége széles skálán mozog, mivel az ultrahangos technológia nem mindig nyújt teljesen tökéletes nyelvkontúrt. A bal felső és jobb alsó képen jól kivehető a nyelv kontúrja; ezzel szemben a bal alsó képen a kontúr nem folytonos, hanem szakadás vagy ugrás látható. A jobb felső képen a nyelvkontúr kevésbé erőteljesen látszik.



2. ábra. Különböző minőségű ultrahangképek ugyanazon beszélőtől.

2.2. A beszédjel előfeldolgozása

A beszédfelvételek és szöveges átiratuk alapján egy magyar nyelvű kényszerített felismerővel [29] meghatároztuk a hanghatárokat, majd a hanghatárok alapján a felvételek elején és végén található csendet nem vettük figyelembe a gépi tanulási adatok generálása során.

A beszédjel paraméterekre bontására és a későbbi visszaállításra egy egyszerű impulzus-zaj gerjesztésű vokódert választottunk (PySPTK implementáció: https://github.com/r9y9/pysptk). Az alapfrekvenciát (F0) a SWIPE algoritmussal mértük. A következő lépésben spektrális elemzést végeztünk mel-általánosított kepsztrum (Mel-Generalized Cepstrum, MGC, [30]) módszerrel, melyet statisztikai parametrikus beszédszintézisben széles körben használnak. Az elemzéshez 25-öd rendű MGC-t számítottunk $\alpha = 0{,}42$ és $\gamma = -1/3$ értékekkel. Ahhoz, hogy a beszédjel analízise során kapott paraméterek szinkronban legyenek az ultrahangképekkel, a kereteltolást 1 / FPS értékre választottuk (ahol FPS az adott ultrahangfelvétel képkocka/másodperc sebessége).

A beszéd visszaállításához az F0 paraméterből először impulzus-zaj gerjesztést generáltunk, majd a gerjesztést és az MGC paramétereket felhasználva MGLSADF szűrővel [31] visszaállítottuk a szintetizált beszédet. A fenti vokóder az SSI témakörében tehát úgy használható, hogy a beszéd visszaállításához az eredeti F0 paraméterek mellett nem az eredeti spektrális paramétereket használjuk fel, hanem az ultrahangképek alapján gépi tanulással becsülteket.

2.3. Az ultrahangadatok előfeldolgozása

Az ultrahangadatokon a csendes szakaszok kivágásán kívül egyéb előfeldolgozást nem végeztünk, azaz közvetlenül az ultrahangos rögzítés során előálló nyers adatok (az 1. ábra középső része) képezték a gépi tanulás inputját, ami gyakorlatilag megfelel annak, mint ha magukon az ultrahangképeken tanítanánk. Így 64 x 842 méretű jellemzővektorokkal kellett dolgoznunk, ami meglehetősen magas jellemzőszámot jelent. A 2.4. fejezetben bemutatunk egy nagyon egyszerű jellemzőkiválasztási módszert, amellyel megpróbáltuk kiszűrni az ultrahangképek azon régióit, ahol nem történik olyan változás, amely a tanulás során fontos lenne a modell számára, így az ide tartozó pixelértékek eldobhatók.

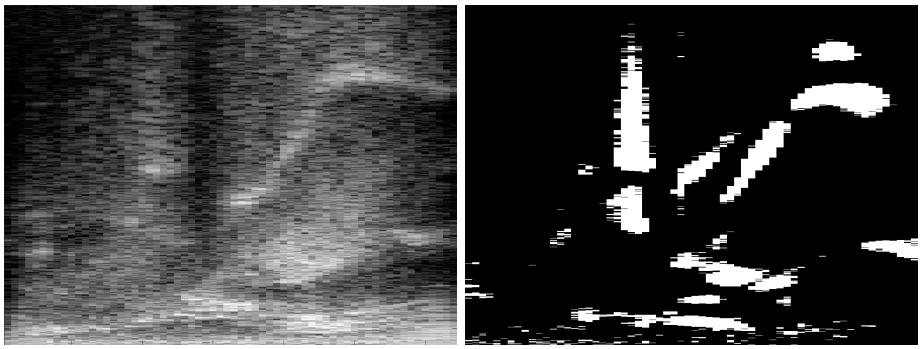
2.4. Gépi tanulás

Az ultrahangfelvételeken teljesen kapcsolt (fully connected) mély ,,egyenirányított" (rectifier) neurális hálókat [32] tanítottunk. A rectifier hálók esetén a rejtett neuronok a rectifier aktivációs függvényt (max(0,x)) alkalmazzák, ennek köszönhetően körülményes előtanítási módszerek nélkül, hagyományos backpropagation algoritmussal is hatékonyan taníthatóak [32]. A megtanulandó célértékeket a vokóder MGC paraméterei képezték. Mivel feltevéseink szerint az utrahangadatokból a hangmagasság értéke (F0) egyáltalán nem, a hangosság értéke (az MGC első dimenziója) pedig csak kis eséllyel állítható vissza, ezért ezt a két paramétert kihagytuk a gépi tanulásból, és a szintézis során

az eredeti értékeket használtuk. A fennmaradó 25 MGC-paraméter a beszéd spektrális burkolóját írja le, a neuronháló feladata ezeknek a paramétereknek a minél pontosabb becslése volt az ultrahang alapján. Mivel ezek a paraméterek folytonos értékűek, ezért osztályozás helyett regressziós módban használtuk a mély hálót. Egyelőre – jobb híján – az átlagos négyzetes hibafüggvény (MSE) segítségével tanítottunk. A későbbiekben érdemes lehet majd ezt leváltani egy olyan mértékre, amely figyelembe veszi az emberi percepciót is. Jaumard-Hakoun és munkatársai például a kiértékelésnél a spektrális torzítást mérték (bár a tanulás során feltehetően ők is az MSE-hibát használták, ez nem derül ki egyértelműen a tanulmányukból) [22]. A multidimenziós regressziós tanítást ők úgy oldották meg, hogy minden regressziós jellemzőre külön neuronhálót tanítottak. Munkánkban mi kipróbáltuk, hogy minden MGC jellemzőre külön hálót tanítva jobb eredményt kapunk-e, mint egy hálót tanítva egyszerre a teljes MGC vektorra.

Kísérleteink során egy 5 rejtett réteges, rétegenként 1000 neuront tartalmazó neuronháló struktúrát használtunk lineáris kimeneti réteggel. Tekintve, hogy az MGC paraméterek különböző skálán mozogtak, tanítás előtt standardizáltuk őket, hogy várható értékük 0, szórásuk pedig 1 legyen. A standardizálás egy fontos lépés, hiszen amennyiben ezt nem tesszük meg, úgy a regressziós tanulás során a nagyobb értékekkel rendelkező MGC jellemzőt tanulja meg a háló nagy pontossággal, míg a kisebb értéktartományon mozgót kevésbé az MSE hibafüggvény miatt.

A neuronhálók bemeneteként kezdetben az egész ultrahangképet használtuk, ami rendkívül zajos, és sok felesleges részt is tartalmaz (lásd 2. ábra), ezért egy egyszerű jellemzőkiválasztási eljárást is kipróbáltunk. A módszer lényege, hogy minden pixelre kiszámítottuk annak korrelációját a 25 MGC jellemzővel, majd vettük ezen korrelációk maximumát, és küszöböltünk, azaz csak azokat a pixeleket tartottuk meg, ahol a korreláció egy küszöbérték fölé esett. A 3. ábra egy példát mutat az eredeti felvételre, illetve a kapott szűrési maszkra (a fehér pontok jelentik a megtartott pixeleket). Az így kapott maszk alapján tudtuk szűrni, hogy a kép mely részeit érdemes figyelni. A bemeneti jellemzőkészlet redukálása



3. ábra. Ultrahangkép és a jellemzőkészlet szűréséhez használt maszk.

révén jelentősen, körülbelül a tized részére – 53 888-ról 5 572-re – redukáltuk a jellemzők számát. Ez a lépés lehetővé tette, hogy ne csak az aktuális ultrahangképet, hanem annak időbeli szomszédait is felhasználjuk a tanítás során. A beszédfelismerésben teljesen szokványos lépés az aktuális adatvektor mellett az időben szomszédos vektorokat is bemenetként megadni a hálónak, innen jött az ötlet erre a megoldásra. A kísérletekben az aktuális képen kívül 4-4 szomszédot használtunk fel inputként, ami összesen 9 szomszédos jellemzővektort jelent; így végső soron a szomszédokat is figyelembe vevő háló nagyságrendileg ugyanakkora inputvektoron dolgozott, mind amekkora az eredeti, redukálatlan inputvektor volt.

3. Kísérleti eredmények

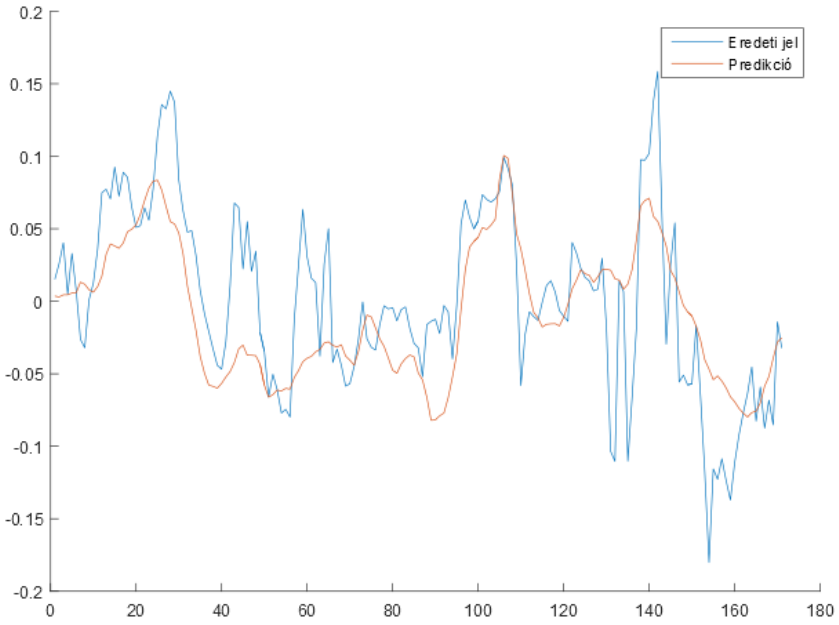
A 176 rendelkezésre álló felvételből 158-at használtunk a neuronhálók tanítására, a maradék 28-at pedig tesztelésre. A neuronháló különböző változataival a teszthalmazon elért átlagos négyzetes hiba (MSE) értékeit az 1. táblázat foglalja össze. A bemeneti jellemzők esetén két variációt próbáltunk meg. „Teljes" jellemzőkészletnek fogjuk hívni azt az esetet, amikor a teljes képet, azaz az összes, 53 888 rögzített adatot használtuk inputként. A korábban ismertetett jellemzőkiválasztási módszerrel előállított 5 572 elemű jellemzőkészletre „redukált" készletként hivatkozunk. A bemeneti képek száma 1 vagy 9 lehet, a 9 jelenti azt, hogy 9 egymást követő kép alkotta az inputot, ami természetesen csakis a redukált jellemzőkészlet esetén jön szóba. A betanított hálók oszlopában az 1-es értékek azt jelentik, hogy egyetlen hálót tanítottunk 25 kimenettel, míg a másik esetben 25 hálót tanítottunk külön-külön a 25 MGC-paraméter becslésére.

A táblázat első és harmadik sorát összevetve láthatjuk, hogy a jellemzők számának radikális csökkentése csak minimális mértékben növelte a hibát, azaz a jellemzőkiválasztási módszerünk jól teljesített. A harmadik és a negyedik sor összevetéséből pedig az olvasható ki, hogy a szomszédos 4-4 kép felhasználása körülbelül 10%-kal csökkentette a hibát. Végezetül, a többi sort is vizsgálva azt látjuk, hogy az egyes paraméterek közelítésére külön-külön tanított hálók nem javítottak számottevően, viszont betanításuk lényegesen több időt vett igénybe.

1. táblázat. A különböző módon tanított neuronhálókkal elért átlagos négyzetes hibák.

Bemeneti jellemzőkészlet	Bemeneti képek száma	Betanított hálók száma	MSE
Teljes	1	1	0,00194
	1	25	0,00190
Redukált	1	1	0,00203
	9	1	**0,00180**
	1	25	0,00199
	9	25	0,00184

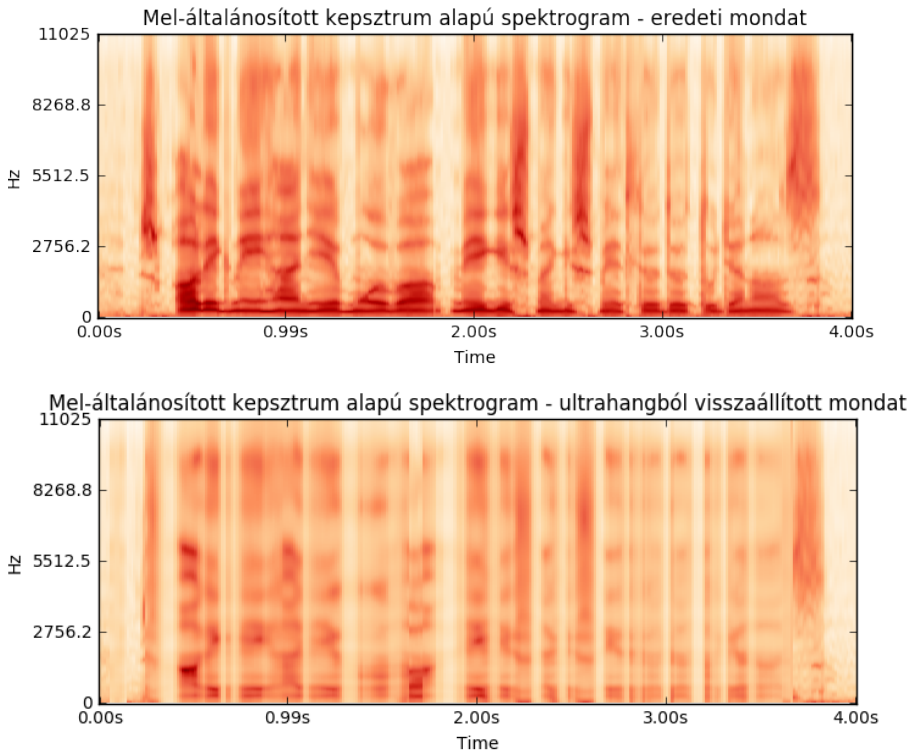
Az MSE hiba értéke sajnos nem túl informatív arra nézve, hogy milyen minőségű lett a visszaállított beszéd. A hiba érzékeltetésére a 4. ábrán kirajzoltuk egy konkrét MGC-paraméter időbeli görbéjét, valamint annak neuronhálóval kapott közelítését. Megfigyelhetjük, hogy a neuronháló alapvetően követi ugyan a görbe trendjét, de a finom részleteket sok esetben képtelen visszaadni. Az ebből eredő hiba csökkentésére tervezzük megvizsgálni, hogy az MGC-paraméterek mekkora időbeli simítást bírnak el minőségromlás nélkül, majd ezekkel a simított paraméterekkel fogjuk tanítani a hálót.



4. ábra. Egy MGC-paraméter időbeli görbéje és annak becslése a legjobb eredményt elérő neuronhálóval.

A hiba további érzékeltetésére az 5. ábrán példát mutatunk egy mondat eredeti, illetve a rekonstrukció után kapott spektrogramjára. Ugyan a neuronháló nem tudta pontosan megtanulni az eredeti beszédre jellemző összes spektrális komponenst (pl. formánsok), de a tendenciák alapján látható, hogy a gépi tanulás eredményeként kapott spektrogram is emlékeztet beszédre (pl. 0,5 s körül a formánsok egészen jól kivehetőek).

Az ultrahangból visszaállított felvételeken precíz, többalanyos lehallgatásos kiértékelést nem végeztünk, de a szubjektív benyomásunk az volt, hogy bár a felvételek nagyon torzak, sok esetben szavak, sőt némely esetben teljes mondatok is érthetőek. Ezt biztató kezdeti eredménynek tartjuk, tekintve, hogy a feldolgozás összes lépésében a lehető legegyszerűbb megoldást alkalmaztuk.



5. ábra. Felül: eredeti MGC-alapú spektrogram. Alul: gépi tanulással artikulációs adatokból becsült MGC-alapú spektrogram.

4. Összefoglalás, következtetések

A tanulmányban bemutattunk egy kísérletet, amelynek a célja az volt, hogy nyelvultrahang-képekből kiindulva beszédet szintetizáljunk. A kutatás során egy női beszélőtől rögzítettünk közel 200 bemondáshoz tartozó szinkronizált beszéd- és nyelvultrahang-felvételt. A beszédből az alapfrekvencia- és a spektrális paramétereket nyertük ki. Ezután mély neurális háló alapú gépi tanulást alkalmaztunk, melynek bemenete a nyelvultrahang volt, kimenete pedig a beszéd spektrális paraméterei. A tesztelés során egy impulzus-zaj gerjesztésű vokódert alkalmaztunk. Az eredeti beszédből származó F0 paraméterrel és a gépi tanulás által becsült spektrális paraméterekkel mondatokat szintetizáltunk. Az így szintetizált beszédben sok esetben szavak, vagy akár teljes mondatok is érthetőek lettek.

A jelen cikkben elért kezdeti eredményeket biztatónak tartjuk. A továbbiakban a rendszernek gyakorlatilag minden pontján finomításokat tervezünk. Meg fogjuk vizsgálni, hogy a szintézis mely paramétereinek becslése a legmegfelelőbb, tervezzük variálni az optimalizálandó célfüggvényt, a ne-

uronháló struktúráját (pl. teljesen kapcsolt helyett konvolúciós), és a jellemzőkinyerési-jellemzőredukciós lépés is rengeteg kísérleti lehetőséget kínál. Emellett a szájpadlás helyzetéről kinyert információ [33] hozzáadása is segítheti a feladat megoldását.

A mai 'Silent Speech Interface' rendszerek ugyan még kísérleti fázisban vannak, de a jövőben várhatóan valós időben is megvalósítható lesz az artikuláció-akusztikum becslés problémája. Az SSI rendszerek hasznosak lehetnek a beszédsérültek kommunikációjában, illetve zajos környezetben történő beszéd során [13]. A beszélőfüggetlen SSI rendszerek elkészítése egyelőre kihívást jelent, de a legújabb kutatások szerint konvolúciós hálózatokkal ebben a témakörben is nagy előrelépést lehet elérni [34].

Az artikuláció és az akusztikum (elsősorban beszéd) kapcsolatának vizsgálata a beszédkutatás alapkérdéseinek megválaszolása mellett hasznos lehet nyelvoktatásban, beszédrehabilitációban, illetve beszédtechnológiában, audiovizuális beszédszintézisben is.

Köszönetnyilvánítás

A kutatás során Csapó Tamás Gábort és Markó Alexandrát az MTA „Lendület" programja; Grósz Tamást az Emberi Erőforrások Minisztériuma ÚNKP-16-3 kódszámú Új Nemzeti Kiválóság Programja támogatta.

Hivatkozások

1. Öhman, S., Stevens, K.: Cineradiographic studies of speech: procedures and objectives. The Journal of the Acoustical Society of America **35** (1963) 1889
2. Bolla, K.: A magyar magánhangzók és rövid mássalhangzók képzési sajátságainak dinamikus kinoröntgenográfiai elemzése. Magyar Fonetikai Füzetek **8**(8) (1981) 5–62
3. Bolla, K., Földi, É., Kincses, G.: A toldalékcső artikulációs folyamatainak számítógépes vizsgálata. Magyar Fonetikai Füzetek **15**(4) (1985) 155–165
4. Stone, M., Sonies, B., Shawker, T., Weiss, G., Nadel, L.: Analysis of real-time ultrasound images of tongue configuration using a grid-digitizing system. Journal of Phonetics **11** (1983) 207–218
5. Stone, M.: A guide to analysing tongue motion from ultrasound images. Clinical Linguistics & Phonetics **19**(6-7) (2005) 455–501
6. Schönle, P.W., Gräbe, K., Wenig, P., Höhne, J., Schrader, J., Conrad, B.: Electromagnetic articulography: use of alternating magnetic fields for tracking movements of multiple points inside and outside the vocal tract. Brain and Language **31**(1) (1987) 26–35
7. Mády, K.: Magyar magánhangzók vizsgálata elektromágneses artikulográffal normál és gyors beszédben. Beszédkutatás 2008 (2008) 52–66
8. Baer, T., Gore, J., Gracco, L., Nye, P.: Analysis of vocal tract shape and dimensions using magnetic resonance imaging: Vowels. The Journal of the Acoustical Society of America **90**(2) (1991) 799–828

9. Woo, J., Murano, E.Z., Stone, M., Prince, J.L.: Reconstruction of high-resolution tongue volumes from MRI. IEEE Transactions on Bio-medical Engineering **59**(12) (2012) 3511–3524
10. Cheah, L.A., Bai, J., Gonzalez, J.A., Ell, S.R., Gilbert, J.M., Moore, R.K., Green, P.D.: A user-centric design of permanent magnetic articulography based assistive speech technology. In: Proc. BioSignals. (2015) 109–116
11. Csapó, T.G., Csopor, D.: Ultrahangos nyelvkontúr követés automatikusan: a mély neuronhálókon alapuló AutoTrace eljárás vizsgálata. Beszédkutatás 2015 (2015) 177–187
12. Hueber, T., Aversano, G., Chollet, G., Denby, B., Dreyfus, G., Oussar, Y., Roussel, P., Stone, M.: Eigentongue feature extraction for an ultrasound-based silent speech interface. In: Proc. ICASSP, Honolulu, HI, USA (2007) 1245–1248
13. Denby, B., Schultz, T., Honda, K., Hueber, T., Gilbert, J.M., Brumberg, J.S.: Silent speech interfaces. Speech Communication **52**(4) (2010) 270–287
14. Bocquelet, F., Hueber, T., Girin, L., Badin, P., Yvert, B.: Robust Articulatory Speech Synthesis using Deep Neural Networks for BCI Applications. In: Proc. Interspeech. (2014) 2288–2292
15. Bocquelet, F., Hueber, T., Girin, L., Savariaux, C., Yvert, B.: Real - time Control of a DNN - based Articulatory Synthesizer for Silent Speech Conversion : a pilot study. In: Proc. Interspeech. (2015) 2405–2409
16. Wang, J., Samal, A., Green, J.: Preliminary Test of a Real-Time, Interactive Silent Speech Interface Based on Electromagnetic Articulograph. In: Proceedings of the 5th Workshop on Speech and Language Processing for Assistive Technologies. (2014) 38–45
17. Denby, B., Stone, M.: Speech synthesis from real time ultrasound images of the tongue. In: Proc. ICASSP, Montreal, Quebec, Canada, IEEE (2004) 685–688
18. Hueber, T., Benaroya, E.L., Chollet, G., Dreyfus, G., Stone, M.: Development of a silent speech interface driven by ultrasound and optical images of the tongue and lips. Speech Communication **52**(4) (2010) 288–300
19. Hueber, T., Benaroya, E.l., Denby, B., Chollet, G.: Statistical Mapping Between Articulatory and Acoustic Data for an Ultrasound-Based Silent Speech Interface. In: Proc. Interspeech, Florence, Italy (2011) 593–596
20. Hueber, T., Bailly, G., Denby, B.: Continuous Articulatory-to-Acoustic Mapping using Phone-based Trajectory HMM for a Silent Speech Interface. In: Proc. Interspeech, Portland, OR, USA (2012) 723–726
21. Hueber, T., Bailly, G.: Statistical conversion of silent articulation into audible speech using full-covariance HMM. Computer Speech and Language **36** (2016) 274–293
22. Jaumard-Hakoun, A., Xu, K., Leboullenger, C., Roussel-Ragot, P., Denby, B.: An Articulatory-Based Singing Voice Synthesis Using Tongue and Lips Imaging. In: Proc. Interspeech. (2016) 1467–1471
23. Gonzalez, J.A., Moore, R.K., Gilbert, J.M., Cheah, L.A., Ell, S., Bai, J.: A silent speech system based on permanent magnet articulography and direct synthesis. Computer Speech and Language **39** (2016) 67–87
24. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. (2015) http://arxiv.org/abs/1506.01497.
25. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105

26. Xie, S., Tu, Z.: Holistically-Nested Edge Detection. In: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE (2015) 1395–1403
27. Olaszy, G.: Precíziós, párhuzamos magyar beszédadatbázis fejlesztése és szolgáltatásai. Beszédkutatás 2013 (2013) 261–270
28. Csapó, T.G., Deme, A., Gráczi, T.E., Markó, A., Varjasi, G.: Szinkronizált beszéd- és nyelvultrahang-felvételek a SonoSpeech rendszerrel. In: Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2017), Szeged, Magyarország (2017)
29. Mihajlik, P., Tüske, Z., Tarján, B., Németh, B., Fegyó, T.: Improved Recognition of Spontaneous Hungarian Speech—Morphological and Acoustic Modeling Techniques for a Less Resourced Task. IEEE Transactions on Audio, Speech, and Language Processing **18**(6) (2010) 1588–1600
30. Tokuda, K., Kobayashi, T., Masuko, T., Imai, S.: Mel-generalized cepstral analysis - a unified approach to speech spectral estimation. In: Proc. ICSLP, Yokohama, Japan (1994) 1043–1046
31. Imai, S., Sumita, K., Furuichi, C.: Mel Log Spectrum Approximation (MLSA) filter for speech synthesis. Electronics and Communications in Japan (Part I: Communications) **66**(2) (1983) 10–18
32. Glorot, X., Bordes, A., Bengio, Y.: Deep Sparse Rectifier Neural Networks. In Gordon, G.J., Dunson, D.B., eds.: Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS). Volume 15., Ft. Lauderdale, FL, USA, Journal of Machine Learning Research - Workshop and Conference Proceedings (2011) 315–323
33. Epstein, M.A., Stone, M.: The tongue stops here: ultrasound imaging of the palate (L). The Journal of the Acoustical Society of America **118**(4) (2005) 2128–31
34. Xu, K., Roussel, P., Csapó, T.G., Denby, B.: Convolutional neural network-based automatic classification of midsagittal tongue gestures using B-mode ultrasound images. submitted manuscript (2016)

A különböző modalitások hozzájárulásának vizsgálata a témairányítás eseteinek osztályozásához a HuComTech korpuszon

Kovács György[1], Váradi Tamás[1]

Magyar Tudományos Akadémia, Nyelvtudományi Intézet,
Budapest VI., Benczúr utca 33.
e-mail:gykovacs@inf.u-szeged.hu, varadi.tamas@nytud.mta.hu

Kivonat Az ember és gép közötti, valamint az emberek közötti interakció fontos kérdése a témairányítás. Gépi felismerésének vizsgálatakor nem csak az érdekes számunkra, hogy milyen pontosság- vagy fedésértékeket tudunk elérni, hanem az is, hogy mely jellemzők mennyiben járultak hozzá ehhez az eredményhez. Kísérleteink során egyéni neuronhálókat tanítottunk a különböző modalitásokból kinyert jellemzők felhasználásával, hogy lemérjük az így kapott neuronhálók teljesítményét a témairányítási címkék osztályozásában. Továbbá megvizsgáltuk, hogy a különböző neuronhálók kimeneteként kapott valószínűség-becslések mely súlyozásával érhetjük el a legjobb osztályozási eredményt. Két modalitás (multimodális, szintaktikai) emelkedett ki a többi közül, a helyes osztályozáshoz való hozzájárulásukkal Az ezen modalitásokból származó jellemzők megfelelő kombinációja ugyanolyan jó eredményt adott, mint az összes modalitás jellemzőinek kombinációja. Továbbá mindkét kombináció jobb eredményt adott mint az összes jellemzőt kombináció nélkül felhasználó neuronháló, sőt ez utóbbi teljesítményét a kizárólag multimodális jellemzőket felhasználó neuronháló is felülmúlta.[1]

Kulcsszavak: HuComTech, témairányítás, valószínűségi mintavételezés, jellemzőkiválasztás

1. Bevezetés

Az ember-számítógép interakció elősegítéséhez fontos, hogy a gép tudja, beszélgetőtársa mikor fejti ki az aktuális témát, mikor tér el attól (kis mértékben módosítva azt, az előzmények figyelembevételével, vagy teljesen eltérve attól), és mikor nem járul hozzá érdemben a témához. Ezért kutatásunk egyik célja, hogy beszélgetés-szegmentumokat témairányítás szempontjából különböző kategóriákba soroljunk. a HuComTech multimodális beszédadatbázisban ezek a kategóriák a következők:

[1] A szerzők köszönetüket fejezik ki az Országos Tudományos Kutatási Alapprogramok (OTKA) programnak, amely a K116938 számú projekt keretében az itt ismertetésre kerülő kutatást támogatta.

- Témakezdeményezés: a beszélő a korábban elhangzottaktól motiváltan új témába kezd, mely illeszkedik a társalgás addigi menetébe.
- Témaváltás: a beszélő oly módon kezd új témába, hogy az a korábbi beszélgetésbe kevésbé illeszkedik, az nem indokolja a téma választását.
- Téma kifejtése: a beszélő az aktuális témát taglalja.
- Hozzájárulás hiánya: szakaszok, melyek nem sorolhatók be egyik korábbi kategóriába sem. Meg kell jegyezzük, hogy ez inkább az egyéb címkék hiánya, mint önálló kategória.

Korábbi cikkünkben [1] kísérleteink többségében követtük ezt a felosztást, azonban jelentősen jobb eredményeket értünk el, amikor a témakezdeményezést (motivált témaváltást) és a motiválatlan témaváltást egyetlen kategóriaként, témaváltásként kezeltük. Ezért jelen cikkünkben ez utóbbi megközelítésre koncentrálunk, és kísérleteink többségében a témairányítási címkék osztályozását három osztály esetére vizsgáljuk.

A témairányítás kérdésköre nem csak az ember és gép közötti kommunikáció elősegítése miatt lehet hasznos, hanem az emberek közötti kommunikáció jobb megértéséhez is. Többek között ez okból nem csak annak lesz jelentősége számunkra, hogy gépi osztályozásában milyen pontosság- vagy fedésértékeket tudunk elérni, hanem hogy mely jellemzők/jellemzőcsoportok járulnak hozzá leginkább az osztályozási eredményekhez. Ezért jelen cikkünkben öt jellemzőcsoportot elemzünk, két különböző módszerrel. Először azt vizsgáljuk, hogy a különböző jellemzőcsoportokokat önmagukban használva milyen eredményeket kapunk, majd azt elemezzük, hogy a jellemzőcsoportok mely kombinációjával kapjuk a legjobb eredményt.

A témában született korábbi munkák főleg lexikális [2,3,4] és prozódiai [5,6] információra, vagy ezek egy kombinációjára támaszkodtak [7,8]. Egyebek mellett a prozódiai információ felhasználását is megvizsgáltuk, ám a lexikális információ közvetlen felhasználására nem volt lehetőségünk, az adatbázis annotációjának sajátosságai miatt. A következő fejezetben az adatbázis bemutatása során ezekről is említést teszünk. Majd az azt követő fejezetben ismertetjük a kísérletek során felhasznált módszereket, miután bemutatjuk az eredményeket, és végül ismertetjük konklúzióinkat, valamint terveinket a jövőbeni munkára.

2. HuComTech multimodális korpusz

A HuComTech projekt keretében 111 beszélővel készült 222 interjú [9]. Minden beszélővel két beszélgetés, egy formális (szimulált állásinterjú), és egy informális beszélgetés került felvételre. A felvételeket aztán az adatforrást tekintve hat modalitás szerint (Multimodális, Szintaktikai, Prozódiai, Unimodális, Videó, Audió) összesen 39 szinten annotálták. Az annotáció elsősorban az interjúalanyra koncentrál, de több olyan eleme is van, amely az interjút készítő viselkedését is leírja. Bár a későbbiekben röviden minden modalitásról szót ejtünk, bővebb leírásra jelen cikk keretei között nincs lehetőség, az adatbázis részletesebb leírása azonban elérhető a projekthez kapcsolódó korábbi publikációkban [9,10,11].

2.1. Modalitások

Az adatbázis annotációja hat modalitásban történt, melyek összesen 221 jellemzőt adtak az osztályozáshoz. A jellemzőket oly módon használtuk, hogy az interjúkat 0,32 másodperces keretekre (frame) bontottuk, és az adott intervallumra jellemző címkét rendeltük az egész kerethez (a bináris jellemzők kivételével ezután az összes jellemzőt 0 átlagra és 1 varianciára standardizáltuk). A különböző modalitásokhoz az alábbi szintek és jellemzők tartoznak:

Multimodális annotáció. Az annotáció ebben a modalitásban a videó és audió adatok együttes felhasználásával készült a Qannot program segítségével. Itt minden információ kétszer jelenik meg: egyszer az interjúalanyra, egyszer az interjút készítőre vonatkozóan. A kategóriából származó információt 29 jellemzőben kódoltuk.

- Kommunikatív aktus: az interjúalany/interjút készítő kommunikatív aktusai, 14 (7-7) bináris (0 vagy 1 értékű) jellemzőben kódolva, a lehetséges címkéknek (none, other, acknowledging, commissive, constative, directive, indirect) megfelelően.
- Támogató aktus: az interjúalany/interjút készítő támogató aktusai, 8 (4-4) bináris jellemzőben kódolva, a lehetséges címkéknek (other, backchannel, politeness marker, repair) megfelelően.
- Témairányítás: az interjút készítő témairányítási aktusai, 3 bináris jellemzőben kódolva, a lehetséges címkéknek (témaváltás, témakezdeményezés, téma kifejtése) megfelelően.
- Információ: azt írja le, hogy az interjúalany/interjút készítő kapott-e olyan információt, amely új volt számára, vagy olyat, amelyet már ismert, esetleg nem kapott semmilyen információt. 4 (2-2) bináris jellemzőben kódoljuk.

Szintaktikai annotáció. A szintaktikai modalitásben egyetlen szint található, melynek 7 mezőjét 20 jellemzőben kódoltuk.

- Clause ID: az aktuális tagmondat helye a mondatban. 1 egész típusú jellemzőben kódolva.
- Alárendeltség: azon tagmondatok azonosítója, melyeknek a jelenlegi tagmondat alá van rendelve, 1 egész típusú jellemzőben (az azonosítók száma) kódolva.
- Egyeztetés: azon tagmondatok azonosítója, melyek egyeztetve vannak a jelenlegi tagmondattal, 1 egész típusú jellemzőben (az azonosítók száma) kódolva.
- Alárendelés: azon tagmondatok azonosítója, melyek a jelenlegi tagmondat alá vannak rendelve, 1 egész típusú jellemzőben (az azonosítók száma) kódolva.
- Beágyazás: azon tagmondatok azonosítója, melyek a jelenlegi tagmondatba ágyazódnak be, 1 bináris jellemzőben kódolva.
- Beágyazódás: azon tagmondatok azonosítója, melyekbe a jelenlegi tagmondat beágyazódik, 1 bináris jellemzőben kódolva.
- Hiányzó kategóriák: a tagmondatból hiányzó kategóriák. 14 bináris jellemzőben kódoljuk, a 14 lehetséges címkének megfelelően.

Prozódiai annotáció. A prozódiai annotáció a Prosotool [12] eszközzel történt. Az ezen modalitásból származó információt 37 jellemzőben kódoltuk.

- F0-mozgás: a simított F0 mozgás az aktuális szegmensben. 5 bináris jellemzőként kódoljuk az öt mozgás-kategóriának (esés, csökkenés, stagnálás, növekedés, emelkedés) megfelelően.
- F0 szint: az alapfrekvencia szintje a jelenlegi szegmens elején és végén. 10 (5-5) bináris jellemzőben kódoljuk, a szegmens elején és végén álló címkék (L_2, L_1, M, H_1, H_2 ahol $L_2 < T_1 < L_1 < T_2 < M < T_3 < H_1 < T_4 < H_2$, és ahol a T_i értékeket küszöbként használjuk) alapján.
- F0 érték: az alapfrekvencia értéke a jelenlegi szegmens elején és végén, 2 valós típusú jellemzőben kódolva.
- Nyers F0 értékek átlaga: az alapfrekvencia értékek átlaga az adott keretre nézve, 1 valós típusú jellemzőben kódolva.
- Zöngés és zöngétlen invervallumok: a megadott intervallum zöngés, zöngétlen (vagy egyik sem), 2 bináris jellemzőben kódolva.
- I-mozgás: az intenzitás változás az adott szegmensben. A jellemzők kódolása ugyan olyan, mint az F0-mozgás esetén
- I-szint: az intenzitás szintje az aktuális szegmens elején és végén. A jellemzők kódolása ugyan olyan, mint az F0 szint esetén.
- I érték: az intezitás értéke az aktuális szegmens elején és végén. A jellemzők kódolása ugyan olyan, mint az F0 érték esetén.

Unimodális annotáció. Ebben a modalitásban az annotáció kizárólag a videó adatok felhasználásával készült, a HuComTech projekt keretében fejlesztett Qannot program segítségével. Az ezen modalitásból származó információt 15 jellemzőben kódoltuk.

- Fordulókezelés: a társalgási fordulók az interjúalany szemszögéből, 5 bináris jellemzőben kódolva.
- Figyelem: leírja, hogy az interjúalany az interjúkészítőre figyel-e, vagy figyelmet vár az interjúkészítőtől, 2 bináris jellemzőben kódolva.
- Egyetértés: az interjúalany által mutatott egyetértés szintje, 7 bináris jellemzőben kódolva.
- Újdonságérték: azt írja le, hogy az interjúalany kapott-e új információt, vagy nem, 1 bináris jellemzőben kódolva.

Videó annotáció. Ebben a modalitasban az annotáció két kategóriában – funkcionális és fizikai)– történt. Amikor az annotátorok a funkcionális szinten dolgoztak (érzelmek és emblémák, a videóhoz tartozó audió jelet is felhasználhatták. A kategóriából származó információt 111 jellemzőben kódoltuk.

- Arckifejezés: a beszélő arckifejezése által tükrözött érzelmek, 7 bináris jellemzőben kódolva.
- Tekintet: a beszélő tekintetének iránya, 6 bináris jellemzőben kódolva.
- Szemöldök: a beszélő szemöldökmozgása, 4 bináris jellemzőben kódolva.

– Fejmozgás: a beszélő fejének mozgása, 8 bináris jellemzőben kódolva.
– Kéz alakja: a beszélő kezei különböző alakzatokat formálhatnak a beszélgetés alatt. Itt ezen alakzatok kerülnek leírásra, 15 bináris jellemzőben kódolva.
– Érintés: annak a leírása, hogy a beszélő melyik kezével, milyen testrészén érintette/vakarta meg magát, 30 bináris jellemzőben kódolva.
– Testtartás: a beszélő testtartásának leírása, 10 bináris jellemzőben kódolva.
– Deixis: a beszélő deiktikus mozgása, 10 bináris jellemzőben kódolva.
– Érzelem: a beszélő látszólagos érzelmi állapota, 7 bináris jellemzőben kódolva. Fontos különbség az arckifejezéshez képest, hogy itt az annotátor az audió csatornát is használhatta a címke kiosztásakor.
– Embléma: a beszélőhöz kapcsolódó embléma címkék (agree, attention, block, disagree, doubt, doubt-shrug, finger-ring, hands-up, more-or-less, number, one-hand-other-hand, other, refusal, surprise-hands), 14 bináris jellemzőben kódolva.

Audió annotáció. Az audió annotáció a tagmondatok szintjén történt. Ez azzal járt, hogy az olyan információkat, mint az egyes szavak, hezitációk, ismétlések, a 25 századmásodpercet meghaladó szünetek, nem tudjuk időben elég pontosan elhelyezni, azaz nem tudjuk ezen jelenségeket a 0,32 másodperces keretekhez kötni. Így az audió annotációból egyedül az érzelmi címkéket használtuk fel, mivel ésszerűen feltételezhetjük, hogy ezek az adott tagmondatra nézve állandóak. Így az audió annotációból első kísérleteinkben egyetlen szintet tudtunk felhasználni, melyet 9 bináris jellemzőben kódoltunk, a megadott címkéknek (silence, overlapping speech, other, happy, neutral, surprised, recalling, sad, tense) megfelelően. Mivel a modalitás címkéinek egyelőre csak töredékét tudtuk jellemzőként hasznosítani, ezt a jellemzőcsoportot jelen cikkünk keretében nem vizsgáltuk.

2.2. Tanító/Validációs/Teszt felbontás

A modellek tanításához, paramétereiknek beállításához valamint a modellek kiértékeléséhez három különálló halmazra van szükségünk: egy tanító-, egy validációs- és egy teszthalmazra. Ezt a felosztást a HuComTech adatbázis esetére 75/10/15 arányban határoztuk meg. Ezt a korábban létrehozott felosztást [1] használtuk jelen munkákban is.

2.3. Az adatok kiegyensúlyozatlansága

A beszélgetések természete miatt sokkal többször fordul elő, hogy kifejtünk egy témát, vagy épp egyáltalán nem járulunk hozzá érdemben egy témához (beszélgetőtársunk viszi a szót) mint az, hogy témát váltunk, vagy új témát kezdeményezünk (a beszélgetések több mint harminc százalékában például egyáltalán nincs motiválatlan témaváltás az interjúalanyok részéről). És az előbbi esetek általában hosszabbak is, mint az utóbbi, ritkább esetek. Így az adatok olyan kiegyensúlyozatlansága lép fel, amely megnehezíti a tanítást és a kiértékelést is. A következő fejezetben leírt módszerekkel többek között erre keresünk megoldást.

3. Kísérleti módszerek

3.1. Súlyozatlan átlagolt fedés

Az osztályok kiegyensúlyozatlan eloszlása problémát jelenthet modelljeink kiértékelésénél. Teszthalmazunkban például az esetek mindössze 18 százaléka tartozik a (motivált vagy motiválatlan) témaváltás kategóriájába, ami azt jelenti, hogy akár 82 százalékos pontosságot is elérhetünk, anélkül, hogy a témaváltásnak akár csak egy esetét is helyesen felismernénk. Ez azt mutatja, hogy a nagyon kiegyensúlyozatlan eloszlású osztályozási feladatok esetén a pontosság nem feltétlenül megbízható mértéke a teljesítménynek. A modellek értékelésének egy népszerűbb mértéke (többek között annak köszönhetően, hogy gyakran használt az Interspeech kihívásokban [13]) a súlyozatlan átlagolt fedés (UAR).

Az UAR az osztályok fedésének súlyozatlan átlaga. Értéke kiszámítható az A tévesztési mátrixból, ahol A_{ij} jelzi j osztály azon elemeit, melyeket az i osztályba soroltunk. Ekkor az UAR értékét a következő képlettel kapjuk:

$$UAR = \frac{1}{N} \sum_{j=1}^{N} \frac{A_{jj}}{\sum_{i=1}^{N} A_{ij}}, \tag{1}$$

ahol N az osztályok száma.

3.2. Valószínűségi mintavételezés

Az osztályok kiegyensúlyozatlan eloszlása a kiértékelés mellett a tanítás során is problémát okozhat. Ha az algoritmusunk egyes osztályokból jelentősen többet lát a tanítás során, mint más osztályokból, az a ritkább osztályok rosszabb felismeréséhez vezethet [14]. Ez olyan extrém módokon nyilvánulhat meg, mint például bizonyos osztályok teljes figyelmen kívül hagyása. Ezt a problémát a különböző osztályokba tartozó elemek számának manipulálásával oldhatjuk meg. Ennek két útja képzelhető el: csökkenthetjük a gyakoribb osztályokba tartozó elemek számát, vagy megpróbálhatjuk növelni a ritkább osztályokba tartozó elemek számát. Az előbbi esetén értékes, nehezen megszerzett tanító adatokat dobunk el, az utóbbi pedig általában csak igen költségesen kivitelezhető. Azonban harmadik lehetőségként manipulálhatjuk úgy az egyes osztályokba tartozó elemek számát, hogy bizonyos elemeket többször felhasználunk a tanítás során. Erre a valószínűségi mintavételezés módszere két lépésben ad lehetőséget. Az első lépésben véletlenszerűen kiválasztjuk az osztályt, majd az adott osztályból véletlenszerűen választunk egy elemet [15]. Az osztály kiválasztását tekinthetjük úgy, mint mintavételt egy multinomiális eloszlásból, feltételezve, hogy minden c_i osztályhoz tartozik egy

$$P(c_i) = \lambda(1/N) + (1 - \lambda)Prior(c_i) \tag{2}$$

valószínűség, ahol N az osztályok száma, $Prior(c_i)$ a c_i osztály a priori valószínűsége, és $\lambda \in [0, 1]$ az eloszlás egyenletességét meghatározó paraméter. Ha $\lambda = 0$, az eredeti eloszlást kapjuk, míg $\lambda = 1$ esetén egyenletes eloszláshoz jutunk [16].

3.3. Mély neuronhálók

Kísérleteinkben egyenirányított mély neuronhálókat alkalmaztunk. Ezek olyan neuronhálók, melyeknek egynél több rejtett rétegük van, és rejtett rétegeikben a neuronok egyenirányítású (rectifier) aktivációs függvényt[2] alkalmaznak a standard szigmoid függvényt helyett. Az elmúlt években jelentősen nőtt ennek az architektúrának a népszerűsége, többek között a beszédfelismerés területén [17]. Az általunk használt neuronhálók három rejtett réteggel készültek, minden rejtett rétegben 250 illetve 1000 neuronnal (attól függően, hogy csak egy adott jellemzőcsoportot, vagy az összes jellemzőt használták bemenetükként). A neuronhálók tanítása a tanító halmazon történt, különböző λ paraméterek és kontextus-méretek mellett. Validációhoz, valamint a tanulási ráta meghatározásához a validációs halmazt használtuk, az UAR értéket használva kiértékelésre.

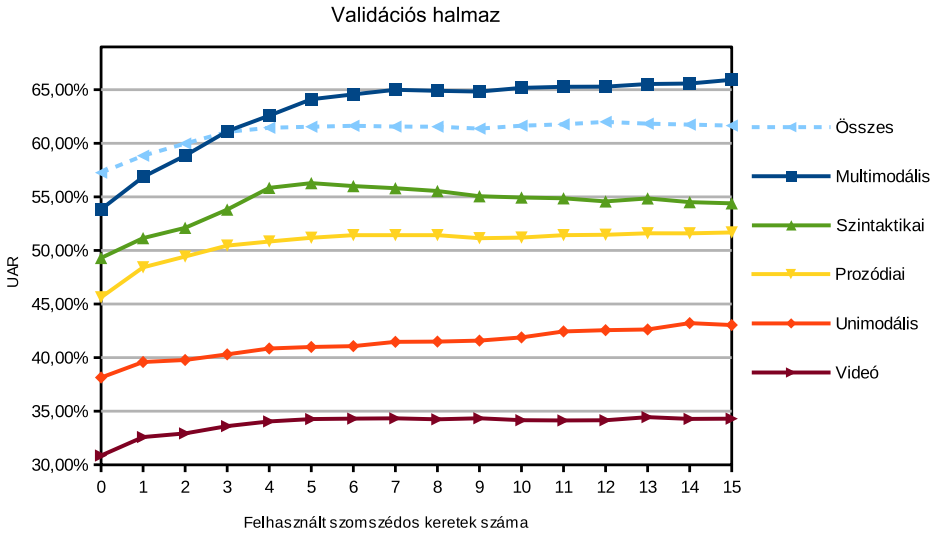
4. Kísérletek egyedülálló jellemzőcsoportokon

Először azt vizsgáltuk, milyen UAR értékeket érhetünk el az egyes jellemzőcsoportok felhasználásával tanított neuronhálók segítségével. Ehhez minden jellemzőcsoporthoz két paramétert kellett meghatároznunk, a bemenetként használt szomszédos keretek számát, valamint a valószínűségi mintavételezésnél használt λ paraméter értékét. Előbbit 0 és 15 (illetve mivel a neuronháló a szomszédokat szimmetrikusan használja, így valójában 0 és 30) között, utóbbit pedig 0 és 1 között (0,1-es lépésközzel) próbáltuk meghatározni. Minden paraméterpárra öt neuronhálót tanítottunk különböző súlyokkal inicializálva, majd megvizsgáltuk, hogy mely paraméterpárra kapjuk a legjobb átlagos UAR értéket a validációs halmazon. A kiértékelést a teszthalmazon ezzel a paraméterpárral végeztük el.

4.1. Eredmények négy osztály esetén

A validációs halmazon kapott eredmények jobb vizualizálása érdekében minden felhasznált szomszédos keretszám esetére kiválasztottuk azt a λ paramétert, amellyel a legjobb UAR eredményt értük el, és ezt az eredményt rendeltük az aktuálisan felhasznált keretszámhoz. Az eredményül kapott diagram a 1. ábrán látható. Az ábráról leolvashatjuk, hogy a különböző jellemzőcsoportok egymáshoz viszonyított teljesítménye meglehetősen stabil. Függetlenül a felhasznált keretek számától, a legjobb eredményt a multimodális jellemzőcsoporttal kapjuk, azt követi a szintaktikai és prozódiai jellemzőcsoport, majd az unimodális jellemzőcsoport, a legrosszabb UAR eredményeket pedig az egyébként legtöbb jellemzőt tartalmazó videó jellemzőcsoport adja. Az egyes jellemzőcsoportok és az összes jellemzőből álló csoport kapcsolata nem ilyen egyértelmű. Amikor a szomszédos kereteket nem használjuk fel a tanítás során, vagy csak keveset használunk közülük, az összes jellemzőt felhasználó neuronháló teljesít a legjobban, ahogy azt várnánk. Három felhasznált szomszédos keret után azonban a multimodális jellemzőcsoporttal jobb eredményeket kapunk.

[2] $recitifer(x) = max(0, x)$

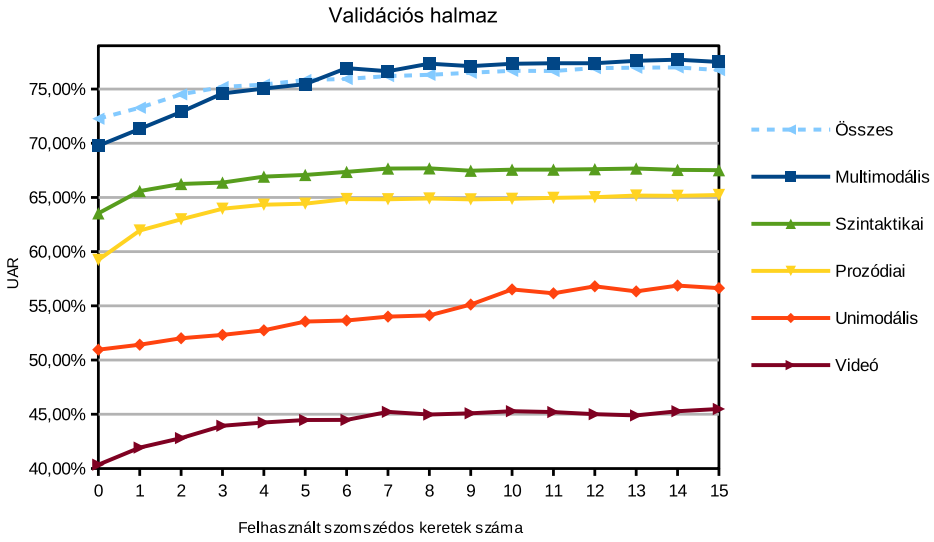


1. ábra. A legjobb elért UAR a különböző jellemzőcsoportokkal a felhasznált szomszédos keretek számának függvényében (öt neuronháló átlaga).

A validációs halmaz alapján minden jellemzőcsoporthoz megtaláltuk azokat a paramétereket, amelyekkel a teszthalmazon kiértékeljük őket. Az így kapott eredmények láthatók az 1. táblázatban. A validációs halmazhoz hasonlóan a teszthalmaz esetén is a multimodális jellemzőcsoport felhasználásával kapjuk a legjobb eredményt, valamint a jellemzőcsoportok sorrendje sem változik. Ám az unimodális és videó jellemzőcsoportok közötti különbség szinte teljesen eltűnik azáltal, hogy az unimodális jellemzőcsoporton tanított neuronhálók eredménye valamelyest romlik a validációs halmazhoz képest, míg a videó jellemzőcsoport eredménye nagy mértékben javul. Az így kapott eredmények továbbra is alacsonyak, ezért további kísérleteinkben a három osztályos esetre koncentrálunk.

1. táblázat. A különböző jellemzőcsoportokon, valamint az összes jellemzőn tanított neuronhálók teszthalmazon történő kiértékelésével kapott UAR eredmények (öt függetlenül tanított neuronháló átlaga).

Jellemző	Szomszédos keretek száma	λ	Validáció	Teszt
Összes	12	1,0	62,0%	62,6%
Multimodális	15	1,0	**65,9%**	**65,0%**
Szintaktikai	5	1,0	56,3%	55,0%
Prozódiai	15	1,0	51,7%	51,5%
Unimodális	14	1,0	43,2%	40,7%
Videó	13	1,0	34,4%	40,5%



2. ábra. A legjobb elért UAR a különböző jellemzőcsoportokkal a felhasznált szomszédos keretek számának függvényében (öt neuronháló átlaga).

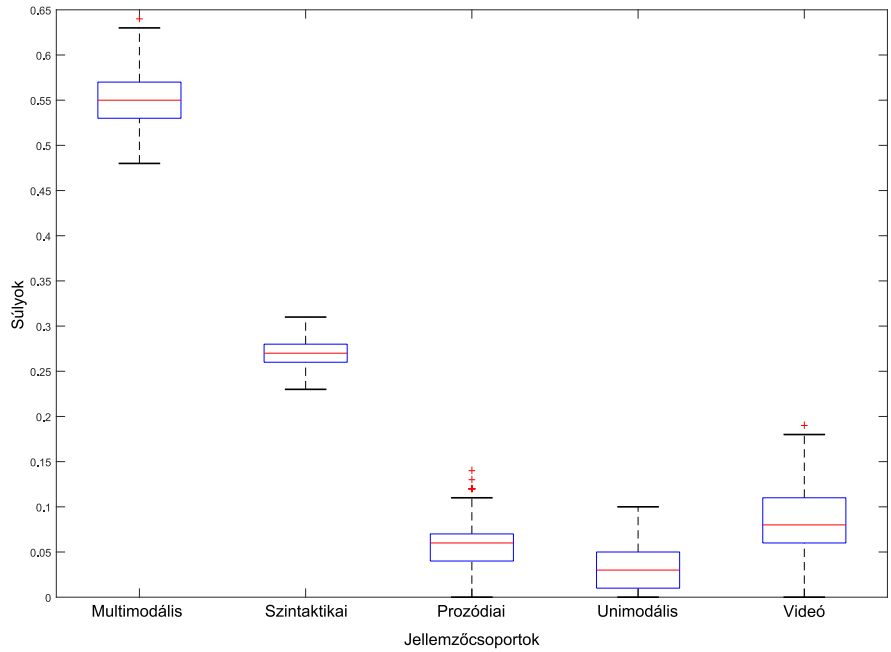
4.2. Eredmények három osztály esetén

A négy osztályra elvégzett kísérleteket megismételtük három osztály esetére. A validációs halmazon kapott eredmények leolvashatók a 2. ábráról. A négyosztályos esethez nagyon hasonló képet látunk: a különböző jellemzőcsoportok teljesítményének sorrendje változatlan, és ismételten azt látjuk, hogy amint a felhasznált szomszédos keretek száma átlép egy korlátot (ezúttal ez 5 keret), egyedül a multimodális jellemzőkkel konzisztensen jobb eredményeket kapunk, mint az összes jellemzővel. Mivel ez esetben a görbék lapultabbak voltak, mint négy osztálynál, a felhasznált szomszédos keretszámot az unimodális jellemzőcsoport alapján állapítottuk meg, 10 szomszédos keretben.

Ismét a validációs halmazon választott paraméterekkel értékeltük ki modelljeinket a teszthalmazon. A 2. táblázatból le tudjuk olvasni, hogy ebben az esetben is a multimodális jellemzőcsoport adta a legjobb eredményt. Láthatjuk továbbá,

2. táblázat. A különböző jellemzőcsoportokon, valamint az összes jellemzőn tanított neuronhálók teszthalmazon történő kiértékelésével kapott UAR eredmények (öt neuronháló átlaga).

Jellemző	Szomszédos keretek száma	λ	Validáció	Teszt
Összes	10	1,0	76,7%	75,7%
Multimodális	10	1,0	**77,3%**	**76,3%**
Szintaktikai	10	0,9	67,6%	67,4%
Prozódiai	10	0,9	64,9%	64,6%
Unimodális	10	0,9	56,5%	55,5%
Videó	10	1,0	45,3%	49,6%

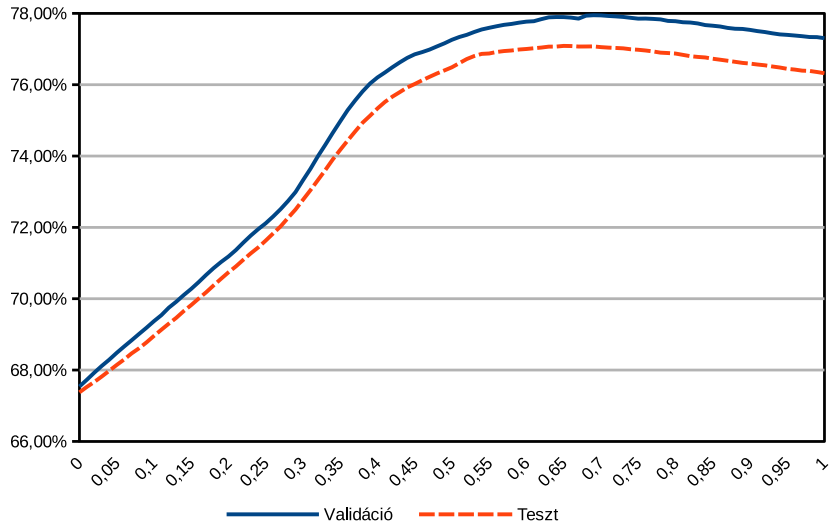


3. ábra. A különböző jellemzőcsoportokhoz tartozó súlyok doboz diagramja.

hogy a jellemzőcsoportok között a validációs halmazon kapott eredmények alapján felállított sorrend ezúttal sem változik a teszthalmaz eredményein.

5. Kísérletek jellemzőcsoportok kombinálására

Mivel a neuronhálók kimeneti rétegében a neuronok softmax függvényt valósítanak meg, így minden neuron kimenete a [0,1] intervallumba esik, és a kimenetek összege 1. Tehát az egyes neuronok kimenetét tekinthetjük az adott osztályba tartozás valószínűségének becsléseként. A különböző jellemzőcsoportokon tanított öt különböző neuronháló tehát öt különböző valószínűségi becslést ad az osztályainkra. A jellemzőcsoportokat úgy próbáljuk kombinálni, hogy ezeknek a valószínűségeknek a súlyozott összegét vesszük, és ez alapján hozunk döntést az osztályozásról. Ehhez 4,5 millió véletlen súlyvektort állítottunk elő 0,01-es lépésközzel, melyeket a validációs halmazon értékeltünk ki, és kiválasztottuk közülük a legjobb UAR eredményre vezető kétezret (itt a legjobb és legrosszabb súlyvektor átlagos teljesítménye között kevesebb, mint 0,05 százalékpontos különbség volt). Ezen kétezer súlyvektorban a különböző jellemzőcsoportokhoz rendelt súlyok terjedelmét, interkvartilis terjedelmét, valamint maximumát, minimumát és mediánját a 3. ábrán ábrázoltuk. Látható, hogy a validációs halmazon legjobban teljesítő súlyozások esetén a legnagyobb súlyokat a multimodális jellemzőcsoport kapta, medián értéke 0,55, míg a szintaktikai jellemzőcsoport medián értéke kevesebb, mint annak a fele (0,27). A súlyok mediánjának sorrendje ettől a ponttól kezdve azonban eltér a jellemzőcsoportok korábbi sor-



4. ábra. UAR eredmények a validációs és teszthalmazon, két jellemzőcsoport esetén a multimodális jellemzőcsoport súlyának függvényében.

rendjétől: a prozódiai jellemzőcsoportot megelőzve, a (korábban legrosszabbul teljesítő) videó jellemzőcsoport következik, és az unimodális jellemzőcsoport zárja a sort.

A 3. ábrán látható, hogy a prozódiai, unimodális és a videó jellemzőcsoport minimális súlya a legjobban teljesítő súlyvektorok között 0. Ezen megfigyelés alapján megvizsgáltuk, milyen UAR eredményeket kaphatunk a validációs halmazon kizárólag a multimodális és a szintaktikai jellemzőcsoportok használatával. Az így kapott eredményeket vizualizálja a 4. ábra. A validációs halmazon akkor kaptuk a legjobb eredményt, ha a multimodális jellemzőcsoport súlya 0,69, a szintaktikai jellemzőcsoport súlya pedig 0,31 volt. Az ezekkel a súlyokkal (2 csoport) kapott eredményt összehasonlítása az összes jellemzőcsoportot használó súlyvektorok közül a validációs halmazon legjobban teljesítő súlyvektorral kapott eredménnyel (5 csoport) és az összes jellemzőt felhasználó (Összes) neuronháló eredményével látható a 3. táblázatban. A két kombináció eredménye között sem a validációs, sem a teszt halmazon nincs szignifikáns különbség, és mindkettő szignifikánsan jobb eredményt ad, mint az összes jellemzőt felhasználó neuronháló magában.

3. táblázat. Az összes jellemzőt felhasználó neuronháló eredményének összehasonlítása a jellemzőcsoportok kombinációjával elért eredményekkel.

Típus	Validáció	Teszt
Összes	76,7%	75,7%
5 csoport	78,1%	77,1%
2 csoport	77,9%	77,1%

6. Konklúzió és jövőbeni munka

Kísérleteink alapján úgy tűnik, hogy a neuronhálós osztályozás eredményességéhez leginkább a multimodális és a szintaktikai jellemzőcsoportok járulnak hozzá. Csak ezen két csoport felhasználásával el tudunk érni az összes csoport kombinációjával kapott eredménnyel egyező eredményt, amely szignifikánsan jobb az összes jellemzőt kombináció nélkül felhasználó eredménynél. A jövőben tervezzük az audió jellemzőcsoport vizsgálatát is, miután a szószintű annotáció rendelkezésünkre áll. Valamint tervezzük, hogy az osztályozási feladatról felismerési feladatra lépünk tovább, HMM/ANN hibrid modell használatával.

Hivatkozások

1. Kovács, Gy., Grósz, T., Váradi, T.: Topical unit classification using deep neural nets and probabilistic sampling. In: Proc. CogInfoCom. (2016) 199–204
2. Sapru, A., Bourlard, H.: Detecting speaker roles and topic changes in multiparty conversations using latent topic models. In: Proc. Interspeech. (2014) 2882–2886
3. Holz, F., Teresniak, S.: Towards automatic detection and tracking of topic change. In: Proc. CICLing. (2010) 327–339
4. Schmidt, A.P., Stone, T.K.M.: Detection of topic change in irc chat logs. http://www.trevorstone.org/school/ircsegmentation.pdf (2013)
5. Baiat, G.E., Szekrényes, I.: Topic change detection based on prosodic cues in unimodal setting. In: Proc. CogInfoCom. (2012) 527–530
6. Zellers, M., Post, B.: Fundamental frequency and other prosodic cues to topic structure. In: Workshop on the Discourse-Prosody Interface. (2009) 377–386
7. Shriberg, E., Stolcke, A., Hakkani-Tür, D., Tür, G.: Prosody-based automatic segmentation of speech into sentences and topics. Speech Commun. **32**(1-2) (2000) 127–154
8. Tür, G., Hakkani-Tür, D.Z., Stolcke, A., Shriberg, E.: Integrating prosodic and lexical cues for automatic topic segmentation. CoRR (2001) 31–57
9. Abuczki, A., Baiat, G.E.: An overview of multimodal corpora, annotation tools and schemes. Argumentum **9** (2013) 86–98
10. Pápay, K., Szeghalmy, S., Szekrényes, I.: Hucomtech multimodal corpus annotation. Argumentum **7** (2011) 330–347
11. Hunyadi, L., Szekrényes, I., Borbély, A., Kiss, H.: Annotation of spoken syntax in relation to prosody and multimodal pragmatics. In: Proc CogInfoCom. (2012) 537–541
12. Szekrényes, I.: Prosotool, a method for automatic annotation of fundamental frequency. In: Proc. CogInfoCom. (2015) 291–296
13. Rosenberg, A.: Classifying skewed data: Importance weighting to optimize average recall. In: Proc. Interspeech. (2012) 2242–2245
14. Lawrence, S., Burns, I., Back, A., Tsoi, A.C., Giles, C.L. In: Neural Network Classification and Prior Class Probabilities. Springer Berlin Heidelberg, Berlin, Heidelberg (1998) 299–313
15. Tóth, L., Kocsor, A.: Training HMM/ANN hybrid speech recognizers by probabilistic sampling. In: Proc. ICANN. (2005) 597–603
16. Grósz, T., Nagy, I.: Document classification with deep rectifier neural networks and probabilistic sampling. In: Proc. TSD. (2014) 108–115
17. Tóth, L.: Phone recognition with deep sparse rectifier neural networks. In: Proc. ICASSP. (May 2013) 6985–6989

Magyar nyelvű WaveNet kísérletek

Zainkó Csaba, Tóth Bálint Pál, Németh Géza,

Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
{zainko, toth.b, nemeth}@tmit.bme.hu

Kivonat: A gépi beszédkeltés legújabb iránya a mély neurális hálózat alapú közvetlen hullámforma generálás. A Google DeepMind kutatói által kidolgozott, ún. nyújtott konvolúció (dilated convolution) alapú WaveNet architektúra képes a hullámforma sajátosságait megtanulni és az így épített modell alapján új hullámformákat generálni. Ezzel az architektúrával magyar adatbázisokon végeztünk kísérleteket. Megvizsgáltuk a hálózat tanulási és generálási képességeit, majd különböző nyelvi jellemzőket felhasználva módosítottuk a tanulási és beszédhullámforma generálási folyamatot. A mondatok generálásához egyrészt természetes bemondásokból kinyert paraméterlistát használtunk, illetve szabály alapú beszédszintetizátor prozódiájával is végeztünk kísérleteket. A generált hangmintákat meghallgatásos teszt segítségével értékeltük, amelyben a WaveNet által generált hangmintákat hasonlítottuk össze természetes és szintetizált beszéddel.

1 Bevezetés

A gépi beszédkeltésnek, a beszédszintézisnek a fejlődését a tudományos eredmények mellett alapvetően meghatározzák az aktuálisan elérhető számítási és tárolási kapacitások. A formáns alapú szintézis a digitális szűrőkön és azok vezérlésén alapul, kihasználva az adott korban elérhető eszközök lehetőségeit [6]. A hullámforma összefűzéses eljárások a 90-es évek elején kezdtek elterjedni, amikor már lehetőség volt a szintézishez szükséges beszédhangminták időtartománybeli reprezentációjának tárolására és futás idejű feldolgozására. Később a háttértárak és memóriakapacitások növekedése lehetőséget nyitott a korpusz alapú beszédszintézisnek, amely akár több gigabájt mennyiségű előre rögzített hangfelvételből válogatja össze a szintetizáláshoz szükséges elemeket. A korpusz alapú beszédszintetizátorokhoz [2] szükséges nagy mennyiségű felvételek lehetővé tették, hogy a döntően szabály alapú gépi megoldások mellett fejlődésnek induljanak a statisztikai elveken működő megoldások [3]. A rejtett Markov-modell (*Hidden Markov Model, HMM*) alapú beszédszintetizátorok már nem a hullámforma szeletekből építik fel a szintetizált beszédet, hanem gépi tanulás útján meghatározott statisztikai paraméterek segítségével beszédkódolót vezérelnek.

1.1 Gépi tanulás alapú beszédszintézis

Már több mint egy évtizede aktívan foglalkoztatja a beszédkutatókat a gépi tanuláson alapuló beszédszintézis [20]. Ezen rendszerekben a beszéd hullámformáját beszédkó-

dolók segítségével paraméterekre bontjuk (alapfrekvencia, spektrális- és időzítési paraméterek), és a szöveges átirat segítségével ezeket a paramétereket tanítjuk be a gépi tanuló modellel. A beszéd generálása során pedig a bemeneti szöveg alapján a modell elkészíti a "legvalószínűbb" paraméter folyamokat, melyekből a beszédkódoló gépi beszédet állít elő.

Korábban rejtett Markov-modell alapú gépi tanulást használtak [15],[20] a paraméterfolyamok modellezésére. Az elmúlt években a HMM-el szemben előtérbe kerültek a nagyobb pontosságot és így jobb beszédminőséget nyújtó mély neurális hálózatok az ugrásszerűen növekedő számítás kapacitásnak – elsősorban a feladatra optimalizált grafikus kártyáknak (*Grapical Processing Unit, GPU*) – és az új tudományos eredményeknek köszönhetően [1], [16], [19].

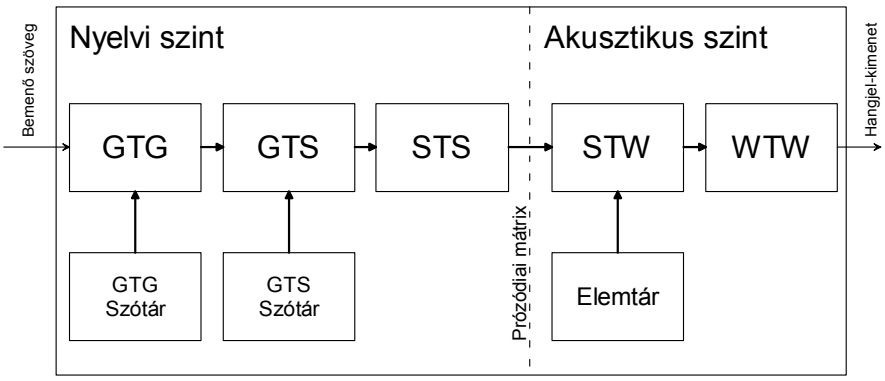
A gépi tanulás alapú beszédszintézisnek számos előnye van a korpusz alapú megoldással szemben: kötetlen témakörben közel azonos minőségű gépi beszédhangot képes nyújtani, kicsi a futásidejű adatbázisa és alkalmas viszonylag rövid (mintegy 10 perc) hangfelvétel alapján a célbeszélő hangjára emlékeztető gépi beszédhangot létrehozni. A beszédkódolók használata azonban hátrányokkal is jár: a paraméterfolyamok nem pontos modellezése esetén a generált beszéd gépiessé, vagy akár hibássá is válhat.

Számos tudományterületen (pl. beszédfelismerés, képosztályozás) a paraméterek analitikus kinyerése (lényegkiemelés) helyett ma már hatékonyan alkalmazzák az ún. mély konvolúciós neurális hálózatokat (*Convolutional Neural Network, CNN*) a paraméterek tanulására [7]. Ez annyit jelent, hogy magukból a nyers adatokból tanulja meg a rendszer, hogy milyen absztrakció írja le legjobban azokat.

Beszédszintézisben először 2016. szeptemberében alkalmaztak a Google DeepMind kutatói CNN-eket a beszéd (és zene) pusztán hullámformából történő modellezésére és generálására. Az új architektúrát WaveNet-nek [11] nevezték el, mely a PixelCNN-ben [13] kidolgozott képgenerálás átültetése hang generálására.

1.2 WaveNet kísérletek

A jelen tanulmányban a WaveNet alapú hullámforma-generáló eljárás magyar nyelvű alkalmazására vonatkozó kísérleteinket mutatjuk be. A WaveNet önmagában nem alkalmas értelmes beszéd szintetizálására, mivel csak a beszédszintetizálás egyik lényeges elemére nyújt megoldást, a hullámforma generálásra, a többire nem.



1. ábra: Példa egy általános TTS felépítésére. A komponensekben használt rövidítések: T: konverzió (to) G: graféma, S: hangkód, W: hullámforma. (az ábra [10] alapján készült)

A beszédszintetizátorok működésének első lépése a nyelvi szint, amit az akusztikai szint követ (1. ábra). A WaveNet az akusztikai szintre ad egy megoldást. Ahhoz, hogy beszédet tudjunk generálni a WaveNet segítségével, a kísérletek során két megoldást használtunk: vagy egy meglévő TTS nyelvi szintjének kimenetét használtuk fel (prozódia mátrix), vagy természetes bemondásból nyertük ezeket a paramétereket.

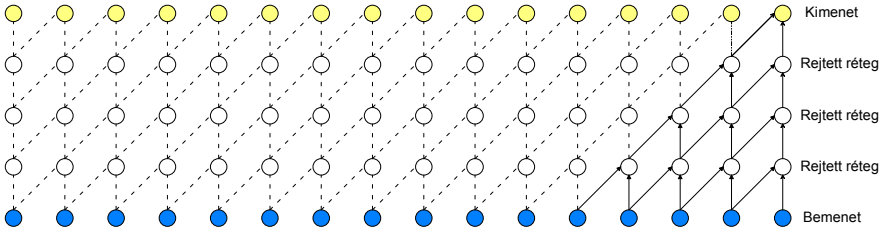
Cikkünk 2. fejezetében ismertetjük a WaveNet hullámformageneráló lényegi elemeit és azok működését. A 3. fejezetben bemutatjuk a kutatás során felhasznált környezetet és beszédadatbázisokat, majd ismertetjük a WaveNet-tel végrehajtott kísérleteinket. A 4. fejezetben bemutatjuk, hogy a kapott modelleket miként értékeltük, és hogy a meghallgatásos tesztek milyen eredményt hoztak. Az utolsó fejezetben pedig összefoglaljuk a tapasztalatainkat és bemutatjuk a továbbfejlesztési irányokat.

2 WaveNet

A WaveNet hálózat kialakítását Oord et al. [12][13] képekre és Józefowicz et al. [5] szövegre alkalmazott megoldásai inspirálták. Azt feltételezték, hogy ha a PixelRNN [13] hálózat képes 64x64 pixeles képeket modellezni, akkor az audió jelek finom struktúráját is lehetséges egy hasonló módszerrel kezelni. A WaveNet kialakításához a PixelCNN-nél [12] is használt felépítést vették alapul, ahol egy képpont generálását a korábbi képpontoktól függő feltételes valószínűségek segítségével adták meg. Az $x=\{x_1,\ldots x_T\}$ hullámformához tartozó feltételes valószínűségeket az (1) képlet adja meg.

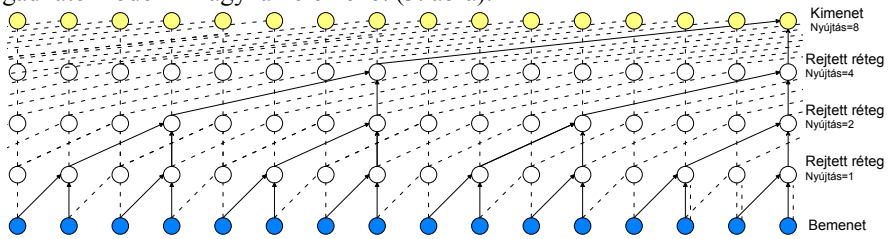
$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1}) \quad \textbf{(1)}$$

Minden x_t minta függ a korábbi időpillanatok mintáitól.



2. ábra: Példa egy 5 rétegű konvolúciós hálózati megoldásra (az ábra [11] alapján készült)

A konvolúciós hálózatban ahhoz, hogy nagyszámú korábbi mintát figyelembe tudjunk venni, nagy számú rejtett réteg, vagy nagy méretű szűrők alkalmazása szükséges (2. ábra). Ezeknek viszont óriásira nőhet a számítási költségük mind a tanítás, mind a generálás során, ezért az ún. nyújtott konvolúciós (*dilated convolution*) architektúrát alkalmazták. Ennek lényege, hogy a rétegek nagyobb részénél, nem az előző időpillanat mintájához tartozó pontokat vonják be a konvolúcióba, hanem paraméterként megadható módon kihagynak elemeket (3. ábra).



3. ábra: Példa egy 5 rétegű nyújtott konvolúciós hálózati megoldásra (az ábra [11] alapján készült)

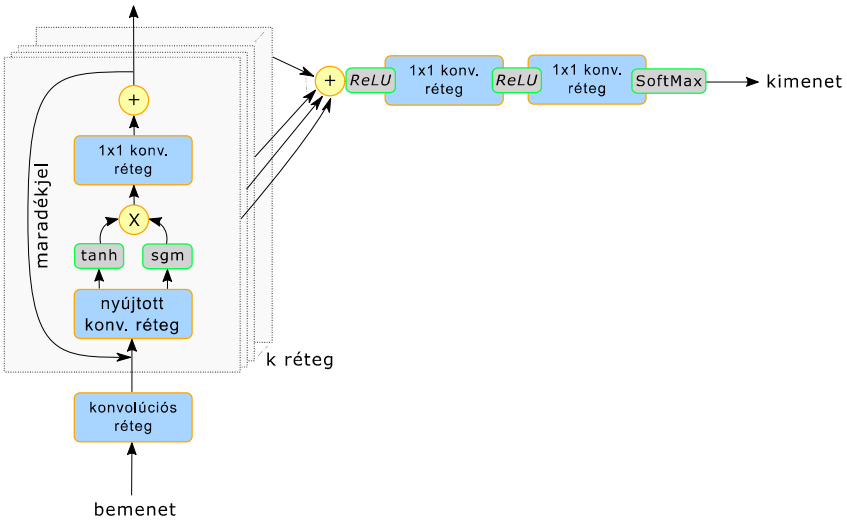
A 2. és a 3. képen is 5 rétegű hálózatot láthatunk. Míg az első hálózat esetében a kimenet az azt megelőző 5 mintától függ, addig nyújtott konvolúció esetén ugyanannyi számítás mellett, 16 mintától függ.

Az audió jelek előállítására tekinthetünk regressziós feladatként, de a digitális jelfeldolgozáshoz és átvitelhez széles körben használt logaritmikus kódolás segítségével osztályozási feladattá lehet átalakítani a problémát. A WaveNet esetében az ITU-T μ-law [4] kódolását használták. A beszédfeldolgozásban tipikusan használt 16 bites lineáris PCM jelet – amely 65536 különböző kvantálási szinttel rendelkezik – átalakítják egy 256 logaritmikus kvantálási szinttel rendelkező μ-law kódolásba. Ezt a 256 szintű reprezentációt utána „one-hot" kódolással adják a hálózat bemenetére, ahol a 256 bemenet közül mindig csak egy tartalmaz nullától eltérő értéket.

A WaveNet esetében ún. kapuzott aktivációs (*gated activation*) egységeket alkalmaznak két aktivációs függvény szorzataként:

$$\mathbf{z} = \tanh\left(W_{f,k} * \mathbf{x}\right) \odot \sigma\left(W_{g,k} * \mathbf{x}\right) \tag{2}$$

A * jelöli a konvolúciót, az x a réteg bemenet, a $W_{f,k}$ a k-dik réteghez tartozó szűrő súlymátrixa, a $W_{g,k}$ pedig a kapu súlymátrixa. A σ a szigmoid függvényt jelöli, a kör pedig az elemenkénti szorzást.



4. ábra: Rétegek kapcsolata egymáshoz és a kimenethez a WaveNet esetében (az ábra [11] alapján készült)

A 4. ábrán látható, hogy a kimenetekhez minden rétegből kivezetjük az adatokat, és azok összegzése és 1x1-es konvolúciója után egy szoftmax függvény adja meg a kimeneti kvantált amplitúdó osztályt.

A WaveNet hálózat ebben a formában csak feltétel nélküli hullámforma generálásra alkalmas. Ahhoz, hogy generáláskor paraméterek segítségével szabályozni tudjuk a generálási folyamatot, több megoldás lehetséges. Az egyik megoldás, hogy a bemenetek mellé párosítjuk a paramétereket. Ezzel a módszerrel végezett kísérleteinket a 3.3-as fejezetben mutatjuk be. A másik módszer – amelyet a Google kutatói is publikáltak [11] – az, hogy a hálózat rétegeibe vezetjük be ezeket az információkat. Ezt a módszert és a kapcsolódó kísérleteinket a 3.4-es fejezetben mutatjuk be.

3 WaveNet kísérletek

A kísérletek elvégzéséhez GPU alapú mély tanuló keretrendszert használtunk. Az adatbázisok előfeldolgozása C++ nyelven történt, a tanítást és a generálást pedig Python alapú TensorFlow (v0.9.0) keretrendszerrel végeztük. A keretrendszer 5.1-es cudnn-t és 7.5-ös CUDA drivert használt. A tanítást GeForce GTX TITAN X-en, a generálást GeForce GTX 970-en végeztük.

3.1 Felhasznált adatbázisok

A különböző tanításokhoz és kísérletekhez egy angol nyelvű több-beszélős és egy illetve több-beszélős magyar nyelvű adatbázisokat használtunk az alábbiak szerint.

VCTK-Corpus [17]: 109 angol anyanyelvű beszélő, beszélőnként kb. 400 felolvasott mondattal. A mondatok főleg újság szövegekből lettek válogatva.

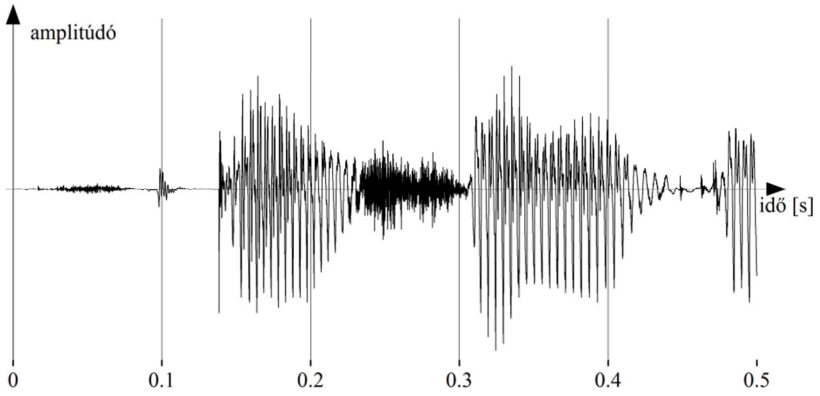
MK_MÁV [18]: Ez a korpusz egy egybeszélős felolvasott korpusz, amely a MÁV állomások hangos utastájékoztató rendszeréhez optimalizált mondatokból áll. A rögzített mondatok stúdióban készültek professzionális rádióbemondó közreműködésével. 3225 mondatot tartalmaz.

MK_RADIÓ [8]: A Nagy et al. [8] által is használt rádiós korpusz. A bemondója megegyezik a MK_MÁV bemondójával. Az adatbázis valós, rádióban elhangzott hírblokkok rögzített hanganyagaiból van összeállítva. Összesen 377 mondatot használtunk fel.

FONETIKA [9]: Egy 2000 mondatos párhuzamos adatbázis, amelyben a hangkapcsolatok fonetikailag kiegyenlítettek. Az adatbázisból 10 beszélőt használtunk, összesen 20000 mondatot.

3.2 Nyers hullámforma előállítás

A WaveNet nyers hullámforma generálásra önmagában is alkalmas, bemeneti minták, címkék vagy feltételek nélkül is. Ekkor nem értelmes szöveget állít elő, hanem hangsorozatokat. Generáláskor a modellt inicializálni kell, induló bemeneti adatként adhatunk valódi beszédmintát, vagy véletlenszerű értékekkel is feltölthetjük a bemenetet. A determinisztikus futás és a reprodukálhatóság miatt a generátort véletlen értékekkel inicializáltuk, de az álvéletlen számgenerátort rögzített értékről indítottuk.



5. ábra: A generált hullámforma időtartományban

A generált minták esetében azt tudtuk vizsgálni, hogy mennyire tartalmaznak beszédhang jellegű részeket (lásd 5. ábra), illetve a hullámforma hangszíne mennyire áll közel a tanító adatbázis beszélőjéhez. A képen látható, hogy a hullámforma a beszédhangokra jellemző képet mutat, különböző zöngés, zöngétlen jellegű szakaszok váltakoznak rajta.

3.3 Bemenet bővítése

A WaveNet bemeneti rétegénél alapesetben a μ-law kódolás eredményeként mintánként 256 különböző bemenet található. Ezeknek a bemeneteknek a száma módosítható, így első lépésként a minták mellé az aktuális beszédhang kódját is beadtuk a hálózatnak, szintén „one-hot" kódolással. A célunk a magyar nyelvű beszédgenerálás, így a további kísérleteket már magyar nyelvű adatbázisokkal végeztük.

Az első kísérletben csak a hullámforma adott részéhez tartozó aktuális hang kódjával bővítettük a bemenetet. A generáláskor azonos módon a hangminták mellé illesztettük a hangkódokat, és így futattuk le a hullámforma generálást. A generált beszéd nem követte a megadott hangkódokat, de megjelent a hangkódok által közvetve megadott időstruktúra. Mivel egy hangkódot annyi mintán keresztül adunk be a hálózat bemenetére, amennyi ideig az adott hang tart, ezért így közvetve hangidőtartam információkat is megadunk.

A bemenetet később 5-ös hangkörnyezetre bővítettük: minden hang esetében az adott hang kódja mellett, az azt megelőző és azt követő két hang kódját is megadtuk. Ekkor már a generált beszédben azonosíthatóak voltak a bemenetre adott hangkódokhoz tartozó hangok.

Az 5 hangos bemeneti kódolást tovább bővítettük a beszéd alapfrekvenciájával. A frekvencia értékek logaritmusát véve osztályokba soroltuk ezt a paramétert és a beszédhang kódokhoz hasonló „one-hot" kódolással vezettük a hálózat bemenetére. Az alapfrekvencia megadása javított a beszéd minőségén, de továbbra is maradtak kevésbé jó minőségű beszédrészletek.

A hálózat bemenetére adott hangkód és alapfrekvencia információk nem elegendőek a megfelelő minőség eléréséhez. Azért választottuk mégis a kezdeti lépésekhez ezt a formát, mert egyrészt a bemenet nagyon egyszerűen bővíthető volt, másrészt így néhány gyorsan kivitelezhető kísérletben meg tudtuk vizsgálni a hangkörnyezet és az alapfrekvencia felhasználhatóságát.

3.4 Mély rétegek szabályozása (*Local Conditon*)

A WaveNet hálózat nem csak a bemeneti rétegen keresztül vezérelhető, hanem minden rétegben módosíthatjuk a szűrő és a kapu súlyok hatását [11]. Az (1)-es képletet módosítva a feltételes eloszlásunk a következő formába írható át:

$$p(\mathbf{x} \mid \mathbf{h}) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1}, \mathbf{h}) \tag{3}$$

A plusz bemeneteket ***h***-val jelöltük. A ***h*** lehet egy globális paraméter, amely hosszú időn keresztül állandó, például a beszélő azonosítója. Amennyiben egy *y=f(h)* függvénnyel a bemeneti mintákhoz illesztett paraméterlistát generálunk (például hangkódok vagy egyéb nyelvi jellemzők), akkor a konvolúciós egységekben lévő aktivációt - (2)-es képlet - a következőképpen módosíthatjuk (ahol $V_{f,k}$*y és $V_{g,k}$*y egy-egy 1x1-es konvolúció):

$$\mathbf{z} = \tanh(W_{f,k} * \mathbf{x} + V_{f,k} * \mathbf{y}) \odot \sigma(W_{g,k} * \mathbf{x} + V_{g,k} * \mathbf{y}) \tag{4}$$

A kísérleteinkben az előző fejezethez hasonlóan a 2-2 beszédhangos környezettel kibővített beszédhangot kódoltuk, illetve az alapfrekvencia logaritmusát.

Több-beszélős adatbázis esetében nem használtuk a $\boldsymbol{h}$ globális paraméterezés lehetőségét, a beszélő azonosítóját is az y bemenetre konvertáltuk át.

3.5 Mondatok generálása

A WaveNet tanítása időigényes, de a nagy mennyiségű feldolgozott adatot figyelembe véve hatékonynak tekinthető. A modell paramétereitől függően egy iterációs lépés kb. 1,4-2 mp-ig tart, amely során 100000 mintát, 6,25 mp hanganyagot használunk fel. A tanítás és a generálás is 16kHz-es mintavételi frekvenciával történt. A hanggenerálás ezzel szemben lassabb művelet, mivel 1 db minta legenerálásához egy teljes forward lépést végre kell hajtanunk, ami alig tart kevesebb ideig, mint egy tanítási lépés. Mivel a következő minta generálásához szükséges az előzőleg generált minta, ezért nem tudjuk a GPU-k párhuzamos számítását kihasználni és egyszerre több mintát előállítani. A mérések szerint egy minta generálása, kb. 0,25 mp-ig tart. Így nagyságrendileg 1 mp hanganyag előállítása kb. 2 óráig tart.

A generálás gyorsítható, amelyre Le Paine [14] adott egy módszert, a Fast-WaveNet-et. A forward lépéseknél olyan számításokat végzünk el minden egyes lépésnél, amit már korábban egyszer kiszámoltunk. La Paine rámutatott, hogy a részeredmények eltárolásával a generálás $O(2^L)$ –ről O(L)-re gyorsítható, ahol L a rejtett rétegek száma. A saját méréseink először nem támasztották alá ezt a gyorsulást. A generálás folyamatát elemezve megállapítottuk, hogy a Fast-WaveNet esetében a numerikus számítás mennyisége annyira lecsökken, hogy a futási idő legnagyobb részét a GPU-ra történő adatátvitel és a számítások után az eredmények memóriába való visszaolvasása adta. Így a GPU-t kihagyva, csak CPU-n futtatva a Fast-WaveNet-et, 1 mp hanganyag előállítása csak kb. 4 percet vett igénybe.

4 Az eredmények értékelése

4.1 Hiba mérése

A neurális hálózatok tanítása során a túltanítás elkerülése céljából a leállási feltételt gyakran a hibafüggvény értékének alakulásához kötjük. Például, ha adott tanítási cikluson keresztül nem csökken a hiba (vagy elkezd növekedni) egy tanító adatoktól elkülönített, ún. validációs halmazon, akkor leállítjuk a tanítást. A WaveNet hálózat tanítása két szempontból is speciális. Egyrészt a generált hangminták esetén jellemző, hogy nem minden esetben a legkisebb hibát produkáló hálózatok adják a legjobb szubjektív értékelést a tesztelőknél. A másik tényező az, hogy a validációs halmaz elemeivel való összehasonlításhoz a hangmintákat le kell generálni. Mivel a generálási idő nagy, egy nagyon kicsinek mondható 2-3 mondatos validációs halmaz legenerálása is 1 teljes napba kerül normál WaveNet generálással (kb. 12 mp hanganyag esetén). Ez alatt a tanítás kb. 50-60 ezer iterációt is el tud végezni, ezért egyelőre inkább a rövid, párhuzamos tesztminták generálása és azok szubjektív értékelése alapján

határoztuk meg, hogy meddig fusson egy tanítás. Fast-WaveNet esetén gyorsabb lenne a validáció, de ezt nem minden hálózaton tudjuk még alkalmazni.

4.2 A szubjektív teszt felépítése

A meghallgatásos teszt három részből állt. Az első részben a tesztelők a magyar nyelvű WaveNet által generált tartalom nélküli hangsorozatot hasonlították össze az eredeti beszélőtől rögzített mondattal. A tesztelőknek azt kellett eldönteniük, hogy menynyire adja vissza a generált hangsorozat az eredeti beszélő hangszínét.

A második részben generált mondatrészleteket kellett összehasonlítani a természetes mondattal. A generált mondatrészekben kétféle módon keltett hangsorozatok szerepelnek. Az első esetben a 3.3-as fejezetben ismertetett módon a bemeneti rétegnél bővítettük a mintákat különböző egyéb információkkal. A mondatok másik részét a 3.4-es fejezetben leírtak szerinti hálózattal generáltuk, ahol a bemeneti paraméterek a rétegek szűrőit és kapuit módosították.

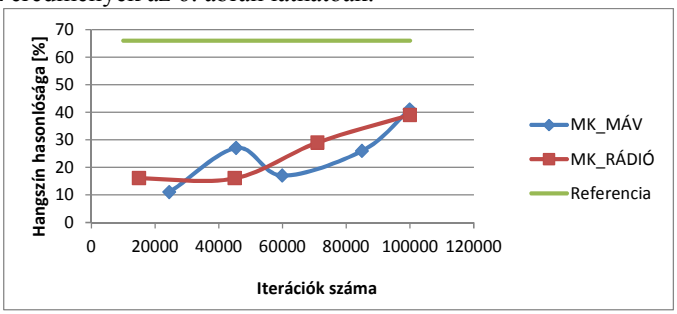
A harmadik részben egy HMM-TTS [8] és egy korpuszos TTS [18] által generált mondatot hasonlítottunk össze a WaveNet MK_MÁV-os valamint a több-beszélős modelljével generált mondattal.

Az értékelét egy internetes MUSHRA (MUlti-Stimulus test with Hidden Reference and Anchor) teszttel végeztük, ahol a résztvevők egy referencia mintához hasonlították a különböző variációkat. A teszt sajátossága, hogy a minták közé a referencia mintát is elhelyeztük, így könnyítve meg a minták skálán való elhelyezését. A tesztelő személy a mintákat egy csúszka segítségével 0-100-as skálán értékelhette, így árnyalatnyinak vélt különbség is megadható volt.

A tesztet 5 nő és 12 férfi végezte el. A legfiatalabb 23 éves volt, a legidősebb 74 éves, az átlag életkor 42 év volt. Valamennyien ép hallásúak és magyar anyanyelvűek.

4.3 A szubjektív teszt eredményei

Az első részben az értelmetlen hangsorozatokat értékelték. A teszt nehézsége az volt, hogy a tesztelőknek a hangszínt kellett értékelni, viszont a szubjektív véleményeket a hangminőség biztosan befolyásolja, csak szakértők tudják ezt megbízhatóan szétválasztani. Az eredmények az 6. ábrán láthatóak.



6. ábra. Az értelmetlen hangsorozatok értékelése két tanító adatbázis esetén.

Az iteráció számával növekedett a hasonlóság a természetes mintához, valószínűleg túl hamar lett leállítva a tanítás, 100000 iteráció után a grafikon alapján még elképzelhető lett volna javulás. Rejtett referenciának a referencia beszéd kevert hangsorozatát raktuk be, amely hangszín szempontjából megegyezik azzal, amihez hasonlították a tesztelők a mintákat, mégis csak 66 %-osra értékelték. Sajnos a vártnak megfelelően nem tudták a hangszínt a minőségtől és értelemtől függetlenül értékelni.

A második részben a különböző módszerrel generált mondatokat hasonlítottuk össze egy természetes mondattal. Az eredmények releváns részei az 1. táblázatban találhatók.

1. táblázat: Generált mondatok minősége összehasonlítva a referenciával

Bemenet	Paraméterek	Iterációk száma	Hasonlóság	Szórás
bem. réteg	1 hang	145k	13 %	3,9
bem. réteg	5 hang	50k	28 %	3,8
bem. réteg	5 hang + logF0	70k	41 %	5,6
mély réteg	5 hang + logF0 + pp	50k	46 %	4,5
mély réteg	5 hang + logF0	55k	53 %	4,8
mély réteg	5 hang + logF0 + pp	200k	54 %	4,6
Referencia			86 %	3,8

Az első oszlop adja meg, hogy a bemeneteket a bemeneti rétegre (3.3 fejezet) vagy a mély rétegekbe (3.4 fejezet) vezetjük be. A bemeneti paraméterek esetében 1 hang, vagy 2-2 hangos környezettel együtt adtuk meg (5 hang). Bizonyos esetekben az alapfrekvencia logaritmusát (logF0) és a prozódiai egységen belüli pozíciót (pp) is megadtuk paraméterként. Az egyhangos bemeneti paraméter egyáltalán nem működött, ezt az eredmények is alátámasztották. A bemeneti rétegre adott plusz információk javítottak a minőségen, de a mély rétegek módosításával jobb eredményeket értünk el.

A teszt harmadik részében korpuszos és HMM technológiával is összehasonlítottuk a generált mondatokat, az eredmények a 2. táblázatban láthatóak.

2. táblázat: Generált mondatok minősége összehasonlítva a referenciával

WaveNet	Korpuszos	HMM	Referencia
40 %	56 %	70 %	81 %

A WaveNet-tel készített mondatot ítélték a leggyengébb minőségűnek, majd a korpusz technológiával készült következett. A korpuszos esetében a szöveg nem a korpusz témakörének megfelelő volt, ezért a mondat minősége rosszabb volt a szokásosnál. A HMM 70 %-ot ért el, a referencia minta pedig 81 %-ot.

5 Összefoglalás

Az első magyar nyelvű WaveNet kísérletek megmutatták, hogy a Google DeepMind kutatói által kidolgozott architektúra alkalmazható magyar nyelvre is. Érthető beszéd már minimális paraméterezéssel is előállítható, 2-2 hangos környezet már elegendő

arra, hogy jól azonosíthatóan megtanulja a hálózat a magyar beszédhangokat. További címkék segítségével javítható a generált beszéd minősége. A meghallgatásos tesztek támpontot adnak a további munkához, de a megbízható értékeléshez több mintát kell generálni, amely most még technológia okokból nehezen kivitelezhető.

Az eljárás legnagyobb hátránya a generálás futásideje, amely nem teszi lehetővé, hogy valós idejű alkalmazásokba ezt a technológiát beintegráljuk.

5.1 Jövőbeli tervek

A beszédminőség javítása érdekében további nyelvi jellemzőkkel érdemes bővíteni a tanításkor és generáláskor használt címkehalmazt. Mivel nem biztos, hogy a több jellemző jobb minőséget eredményez, ezért ezen címkék optimális halmazának kiválasztása az egyik cél.

Mivel a generálási idő komoly korlátot jelent a felhasználás szempontjából, ezért a sebesség növelése a másik prioritás a jövőben. Ahhoz, hogy széles körben használható legyen, legalább valós idejű működés szükséges, amely azt jelenti, hogy a most használt Fast-WaveNet generálási módszernél is legalább 240-szer gyorsabb eljárás szükséges.

A tesztmondatok generálása során azt tapasztaltuk, hogy a betanított modellek minőségén túl a generálás inicializálása is jelentősen befolyásolja a generált beszéd minőségét. A további kutatásaink során ezzel a részterülettel is behatóbban szeretnénk foglalkozni.

5.2 Köszönetnyilvánítás

Köszönjük Bartalis István Mátyásnak a meghallgatásos teszt létrehozásában nyújtott segítségét. Tóth Bálint Pál köszöni az NVIDIA vállalat támogatását, a kutatási célokra rendelkezésére bocsájtott NVidia Titan X GPU kártyát.

A generált magyar nyelvű WaveNet minták a http://smartlab.tmit.bme.hu/wavenet oldalon meghallgathatók.

Bibliográfia

1. Fan, Y., Qian, Y., Xie, F. L., & Soong, F. K. TTS synthesis with bidirectional LSTM based recurrent neural networks. In Interspeech (2014). pp. 1964-1968.
2. Fék M, Pesti P, Németh G, Zainkó C, Olaszy G. Corpus-based unit selection TTS for Hungarian. In: International Conference on Text, Speech and Dialogue (2006 Sep 11) pp. 367-373. Springer
3. Heiga, Zen., et al. "Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005." IEICE transactions on information and systems 90.1 (2007): 325-333.
4. ITU-T. Recommendation G. 711. Pulse Code Modulation (PCM) of voice frequencies, (1988)

5. Jozefowicz R, Vinyals O, Schuster M, Shazeer N, Wu Y. Exploring the limits of language modeling. arXiv preprint arXiv:1602.02410. (2016 Feb 7).
6. Klatt, Dennis H. "Review of text-to-speech conversion for English." The Journal of the Acoustical Society of America 82.3 (1987): 737-793.
7. Le Cun, Y., & Bengio, Y. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10), (1995)
8. Nagy, Péter, Csaba Zainkó, and Géza Németh. "Synthesis of speaking styles with corpus-and HMM-based approaches." Cognitive Infocommunications (CogInfoCom), (2015) 6th IEEE International Conference on. IEEE.
9. Olaszy, G., "Precíziós, párhuzamos magyar beszédadatbázis fejlesztése és szolgáltatásai Beszédkutatás (2013), pp. 261–270, 2013.
10. Olaszy, G., Németh G., Olaszi, P., Kiss, G., Gordos , G.: "PROFIVOX - A Hungarian Professional TTS System for Telecommunications Applications", International Journal of Speech Technology, Volume 3, Numbers ¾ , (December 2000), pp. 201-216.
11. Oord AV, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K. WaveNet: A generative model for raw audio. arXiv preprint arXiv:1609.03499. (2016 Sep 12.)
12. van den Oord A, Kalchbrenner N, Kavukcuoglu K. Pixel Recurrent Neural Networks. arXiv preprint arXiv:1601.06759. (2016 Jan).
13. Oord AV, Kalchbrenner N, Vinyals O, Espeholt L, Graves A, Kavukcuoglu K. Conditional image generation with pixelcnn decoders. arXiv preprint arXiv:1606.05328. (2016 Jun 16.)
14. Tom Le Paine: Fast Wavenet: An efficient Wavenet generation implementation https://github.com/tomlepaine/fast-wavenet (2016.nov.10)
15. Tóth Bálint Pál, Németh Géza, Rejtett Markov-modell alapú szövegfelolvasó adaptációja félig spontán magyar beszéddel, In: VI. Magyar Számítógépes Nyelvészeti Konferencia], Szeged, Magyarország, (2009), pp. 246-256
16. Wu, Z., Valentini-Botinhao, C., Watts, O., & King, S.. Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2015, April) pp. 4460-4464. IEEE.
17. Yamagishi, Junichi. English multi-speaker corpus for CSTR voice cloning toolkit, (2012). URL http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html
18. Zainkó, Csaba, et al. "A Polyglot Domain Optimised Text-To-Speech System for Railway Station Announcements." Sixteenth Annual Conference of the International Speech Communication Association. (2015).
19. Zen, H., Senior, A., & Schuster, M. Statistical parametric speech synthesis using deep neural networks. In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (2013, May). pp. 7962-7966. IEEE.
20. Zen, H., Tokuda, K., & Black, A. W. Statistical parametric speech synthesis. Speech Communication, 51(11), (2009). 1039-1064

IV. Szentimentelemzés

A kognitív disszonancia narratív markereinek azonosítása termékleírásokban

Pólya Tibor

Magyar Tudományos Akadémia, Természettudományi Kutatóközpont,
Kognitív Idegtudományi és Pszichológiai Intézet, Pf.: 286
1519 Budapest, Magyarország
{polya.tibor@ttk.mta.hu}

A tanulmányban bemutatott vizsgálat célja a termékleírást végző személy kognitív disszonanciájának mértékét jelző narratív markerek azonosítása. A kognitív disszonancia lehetséges narratív markereit a kognitív disszonancia szociálpszichológiai elméletei és a termékleírások történetszerű szerveződésére vonatkozó elképzelés alapján fogalmaztam meg. Az empirikus teszteléshez internetes véleménygyűjteményben elérhető autóleírások szövegét elemeztem a NarrCat eljárással. A termékleírások szövegének elemzése számos narratív markert azonosított. Az elemzés értékelését automatikus klasszifikációs eljárással végeztem el. Ennek eredményei azt mutatják, hogy narratív markerek alapján 80 % fölötti megbízhatósággal azonosítható a termékleírást készítő személy kognitív disszonanciájának mértéke.

1 Bevezetés

A számítógépes nyelvészeti fejlesztések egyik legdinamikusabban fejlődő területe a szentimentelemzés. A szentimentelemzés során annak megállapítása a cél, hogy valamely entitás értékelése pozitív vagy negatív az elemzett szövegben [9]. A szentimentelemzést azonban sok esetben hasznos lehet kiterjeszteni a szöveget létrehozó személy vizsgálatára, azaz arra a kérdésre keresni a választ, hogy az entitást hogyan értékeli a személy. Különösen fontos lehet ez a kiegészítés a termékleírások, illetve termékértékelések esetén, mivel ebben az esetben az is fontos lehet, hogy pontosabban értsük meg, illetve predikáljuk a vásárlóként megjelenő személy viselkedését. A szentimentelemzés feladatának ez a kiterjesztése azonban számos kihívást is magában foglal. Többek között azért, mert egy termék értékelését a termékkel kapcsolatos tapasztalatok mellett a személy aktuális motiváció állapota is befolyásolhatja. Jelen tanulmány az értékelést adó személy pszichológiai állapotának egy összetevőjét, a kognitív disszonancia jelenlétét vizsgálja. Elsősorban arra vagyunk kíváncsiak, hogy fel tudjuk-e ismerni azokat a termékleírásokat, illetve értékeléseket, amelyek a kognitív disszonancia állapotában keletkeztek. A kognitív disszonancia jelenséget és a magyarázó elméletek közül azokat, amelyek vizsgálatunk szempontjából fontosak, az első fejezetben mutatjuk be.

A szentimentelemzés irodalma számos fogalmat használ a szentiment meghatározására, úgymint érzés, érzelem, attitűd és vélemény. Ezek közül a fogalmak közül az attitűd fogalmát fogom használni. Választásomat a következő négy indokkal támasztom alá. Az attitűd fogalma összegzett értékelést jelent, így az attitűd fogalma a szentiment fogalmának jelentős részét lefedi. Az attitűd szintén összetett fogalom, hiszen az attitűdöknek van kognitív, érzelmi és viselkedéses összetevője is, következésképp a szentiment fogalmában meglévő különbségeket is képes lefedni az attitűd fogalma. A vizsgálatom középpontjában álló kognitív disszonancia szintén az attitűd fogalmához kapcsolódó jelenség. Végül az attitűdre és azon belül a kognitív disszonanciára vonatkozóan is rengeteg empirikus kutatási eredmény áll rendelkezésünkre ezen jelenségek intenzív szociálpszichológiai kutatásának köszönhetően.

2 A kognitív disszonancia jelensége és magyarázata

Az emberek alapvetően motiváltak arra, hogy úgy gondolhassák, helyes döntést hoztak. Szociálpszichológiai szempontból egy termék megvásárlása szintén döntésnek tekinthető. A termékleírás vagy termékértékelés alkotója így rendszerint motivált arra, hogy úgy gondolhassa, hogy a legjobb terméket vásárolta meg. Az önigazolás motivációjának kielégülését azonban nagymértékben nehezíti az, hogy hétköznapi helyzetekben meghozott döntéseink során nem egy minden szempontból jó, illetve egy minden szempontból rossz alternatíva közül kell választanunk. A legtöbb esetben a választásai alternatíváink bizonyos szempontból vagy szempontokból jónak, más szempontból vagy szempontokból azonban rossznak tekinthetők. A hétköznapi helyzetekben meghozott döntéseink során így azt kell mérlegelni, hogy az egyes alternatívák milyen mértékben tekinthetők jónak, illetve rossznak. Mindebből az is következik, hogy a választott termék, bár jó esetben jelentős mennyiségű jó tulajdonsággal bír, emellett mégiscsak tartalmaz valamennyi rossz tulajdonságot is. A nem választott termék pedig amellett, hogy jelentős mennyiségű rossz tulajdonsággal bír, valamilyen mértékben jó tulajdonságokkal is rendelkezik. Amennyiben ez így van, akkor elmondható, hogy a hétköznapi választásaink során, amikor például két termék közül kiválasztjuk, hogy melyiket vásároljuk meg, ez a választásunk implikálni fog bizonyos mértékű rossz tulajdonság választását és bizonyos mértékű jó tulajdonság elutasítását. Ez a következmény ütközik az önigazolásra törekvés motivációjával, és így a választásainkat követően gyakran megjelenik a kognitív disszonancia állapota. A kognitív disszonancia egy kellemetlen állapot, amely arra motiválja a személyt, hogy csökkentse a disszonancia mértékét. A kognitív disszonancia csökkentésének számos módja van. Témánk szempontjából a legfontosabb az, hogy ha a személy úgy tudja látni, hogy az általa megvásárolt termék több jó tulajdonsággal rendelkezik, illetve kevesebb rossz tulajdonsággal rendelkezik, akkor csökken a kognitív disszonancia mértéke. A kognitív disszonancia állapotában készített termékleírás tehát azért mutathat pozitívabb értékelést, hogy a kognitív disszonancia mértékét ezáltal csökkentse a személy. Ezt a helyzetet fontos lehet megkülönböztetni attól a helyzettől, amikor a személy a termékkel való jó tapasztalataira építve értékeli pozitívan a terméket.

Fontos kiemelni, hogy a kognitív disszonancia által motivált értékelés nem jelent hazugságot. Nem arról van szó, hogy a személy látva a termék negatív tulajdonságait szándékosan torzítja az értékelését pozitív irányba. Ehelyett arról van szó, hogy a személy őszintén meg van győződve a termék pozitív tulajdonságairól, és nincs tudatos rálátása arra, hogy honnan ered a termék pozitív értékelése. Sok esetben az emberek akkor is vonakodnak elismerni, hogy pozitív értékelésük mögött a kognitív disszonancia állapota van, ha a szociálpszichológusok beavatják őket a kognitív disszonancia működésébe.

A kognitív disszonancia jelenségét elsőként Festinger [6] írta le és az első magyarázó elméletet is ő dolgozta ki. Festinger magyarázata szerint kognitív disszonancia akkor keletkezik, amikor az ember felismeri, hogy van két olyan gondolata, amely között logikai ellentmondás van abban az értelemben, hogy az egyikből a másik ellentéte következik. Amennyiben a személy arra gondol, hogy az általa vásárolt termék jó és rossz tulajdonságokkal is rendelkezik, rendszerint megjelenik a kognitív disszonancia állapota [3].

A Festinger által kidolgozott kognitív disszonancia elméletnek számos riválisa elmélete is van. Ezek közül háromról ejtek szót, mivel egyrészt ezek az elméletek a termékleírás készítés helyzetének fontos jellemzőire hívják fel a figyelmet, másrészt a kognitív disszonancia narratív markereinek azonosításához is kiindulási alapot adnak ezek az elméletek.

Aronson [1] magyarázata az én érintettségének szempontját emeli ki a kognitív disszonancia állapotának keletkezésében. A magyarázat szerint a kognitív disszonancia keletkezéséhez nem elegendő a tudattartalmak közötti logikai ellentmondás, az énnek is involváltnak kell lennie az ellentmondásban. A vásárlás után előálló kognitív disszonancia állapotára vonatkozóan ez azt jelenti, hogy az „általam választott termék jó tulajdonságokkal bír” és „az általam választott termék rossz tulajdonságokkal is bír” gondolatok közötti ellentmondás az, aminek pszichológiai jelentősége van.

Tedeschi és munkatársai [12] magyarázata a cselekvő személy által másokban keltett benyomások szerepét hangsúlyozza a kognitív disszonancia állapotának keletkezésében. Ez a magyarázat nem a gondolatok közötti logikai ellentmondásból eredezteti a kognitív disszonancia állapotát. Ehelyett arra helyezi a hangsúlyt, hogy a cselekvéseknek és a cselekvések kiváltó okainak a cselekvések racionalitását megítélő más személyek számára kell konzisztensnek lenniük. Ez a magyarázat azért jelentős vizsgálatunk szempontjából, mert az internetes termékleírások kifejezetten mások számára készülnek és a vásárlások racionalitásának külső személy általi megítélése hangsúlyosan van jelen ebben a helyzetben.

Végül Billig [2] magyarázata azt emeli ki, hogy a társas kontextus további módon is befolyásolhatja a kognitív disszonancia állapotának keletkezését. Billig szerint a kognitív disszonancia állapota előállhat úgy is, hogy a gondolatok és értékelések közötti ellentmondás nem egy személyen belül, hanem két személy között jelentkezik. A kognitív disszonancia állapot tehát a magyarázat szerint akkor is jelentkezhet, amikor a vásárló úgy gondolja, hogy számos jó tulajdonsággal rendelkezik az általa vásárolt termék, de személynek tudomása van arról, hogy mások szerint jóval kevesebb a termék jó tulajdonságainak száma. Billig szerint a kognitív disszonancia állapota úgy szüntethető meg, ha sikerül a saját és a másik személy gondolattartalmait ellentmondás mentesen összeegyeztetni.

Összefoglalásképpen elmondható, hogy a kognitív disszonancia állapota a vásárlások után nagy valószínűséggel előálló állapot. Az állapot keletkezése számos forrásból eredhet és a gondolkodást és a viselkedést jelentősen befolyásoló motiváló szereppel bír.

3 A kognitív disszonancia narratív markerei

A kognitív disszonancia állapotának motivációs szerepe miatt feltételezhető, hogy az állapot fennállása a termékleírások szövegének megformálására is hatással van. A kognitív disszonancia állapotával összefüggésben levő narratív markerek azonosításához két módon kerülhetünk közelebb. Egyrészt a kognitív disszonancia korábban röviden bemutatott szociálpszichológiai magyarázatai alapján azonosíthatjuk, hogy a termékleírás megformálásában milyen különbségek jelentkezhetnek a magas és az alacsony mértékű kognitív disszonancia mellett. Másrészt azt vizsgálom meg, hogy a kognitív disszonancia állapotának fennállása hogyan befolyásolja a termékkel kapcsolatos tapasztalatokról és értékelésekről beszámoló szövegek történetszerűségének mértékét.

3.1 Hipotézisek

A hipotézisek arra vonatkoznak, hogy milyen különbségek várhatóak a kognitív disszonancia magas és alacsony mértékű állapotában írt termékleírások megformálásában. Az első feltevésünk az, hogy a személy által adott értékelés mindkét esetben pozitív. Így azt várom, hogy sem az értékelések számában, sem azon belül a pozitív és negatív értékelések számában nem lesz különbség a termékleírások két csoportja között. A szentiment kifejezésére az érzelmek is képesek. Az érzelmek esetében az előzőekkel megegyezően azt várom, hogy sem az érzelmi kategóriák előfordulásban, sem azon belül a pozitív és negatív érzelmek számában nem lesz különbség a termékleírások két csoportja között.

Emellett azonban számos különbség megjelenésére is számíthatunk a termékleírások szövegének megformálásában. Festinger magyarázata szerint a kognitív disszonancia a tudattartalmak logikai ütközéséből ered. A tudattartalmak fontos szerepe a kognitív disszonancia jelenségében azt eredményezheti, hogy a kognitív disszonancia magas mértékű állapotában elkészített termékleírásokban gyakrabban jelennek meg a termékleírást készítő, illetve más személyek gondolatai, mint az alacsonyabb kognitív disszonancia állapotában készült termékleírásokban.

Aronson magyarázata szerint kognitív disszonancia megjelenéséhez az én érintettsége is szükséges. Az én fontos szerepe azt eredményezheti, hogy az én gyakrabban jelenik meg a magas mértékű kognitív disszonancia állapotában írt termékleírásokban, mint az alacsonyabb mértékű kognitív disszonancia állapotában készült leírásokban.

Tedeschi és munkatársai, illetve Billig magyarázataiból következően a kognitív disszonancia keletkezésében fontos szerepe van más személyeknek, ami azt eredményezheti, hogy a magas mértékű kognitív disszonancia állapotában írt termékleírások-

ban gyakrabban jelenik meg a személy kategória, mint az alacsony mértékű kognitív disszonancia állapotában készült termékleírásokban.

A történetkonstrukció rendszerint a múltbeli eseményekre időben visszatekintő pozícióból történik [4]. Mivel a kognitív disszonancia állapota erős motivációs erővel bír, az ebben az állapotban lévő személy az aktuális helyzetére fókuszál, ami nem kedvez az állapotra való rátekintésnek. A történetkonstruálás akkor kezdődhet el, amikor a kognitív disszonancia állapota megszűnik. A személy ekkor kerül megfelelő pszichológiai távolságra a disszonancia állapotától a visszatekintéshez. A termékleírások történetté szerveződését a téridői perspektíva mutatja [10]. A magas mértékű kognitív disszonancia állapota a metanarratív és az átélő perspektíva formák használatának kedvez, így azt várom, hogy e két perspektíva forma használata gyakoribb, a visszatekintő perspektíva formáé pedig ritkább lesz ezekben a termékleírásokban, összehasonlítva az alacsony mértékű kognitív disszonancia állapotában írt termékleírások szövegével.

4 Vizsgálat

4.1 Szövegkorpusz

A szövegkorpuszt a Népítélet nevű honlapon (www.nepitelet.hu) szereplő autóleírásokból állítottam össze. A honlap célja, hogy fórumot adjon az emberek autókkal szerzett tapasztalatainak és értékeléseinek megosztásához. A leírások közül a Fiat Multipla típusát választottam ki, mivel feltételezhető, hogy az autó szokatlan formája felerősíti a kognitív disszonancia mértékét. A Multipla tulajdonosoknak pozitívabbnak kell látniuk az autójuk más tulajdonságait, hogy ellensúlyozzák a jellemzően negatívan megítélt szokatlan forma hatását. A honlapon 54 Multipla leírást találtam. A szövegkorpusz emellett tartalmazza 4 olyan autótípus leírásait is, amelyek megszokott formája vélhetően nem növeli a kognitív disszonancia szintjét és adott esetben akár reális alternatívái is lehetnek a Multipla választásának (Ford Mondeo, Toyota Avensis, Renault Laguna és Volkswagen Passat). A kiválasztott autók értékét egyeztettem a Használtautó adatbázis (www.hasznaltauto.hu) alapján. Mind a 4 autótípus esetében 20-20 leírást választottam ki véletlenszerűen. Mivel rövid szövegekre nem jellemző a történetszerű szerveződés, az elemzésből kihagytam a rövid, 30 szónál rövidebb leírásokat (5 leírás). Az elemzésbe bevont szövegkorpusz 129 szöveget tartalmaz. A szövegkorpusz 52 Multipla és 77 más autótípus leírását tartalmazza. Az autóleírások teljes terjedelme 24 514 szó.

4.2 Elemzés

Az autóleírások szövegének elemzéséhez a Narratív Kategoriális Tartalomelemzőt használtam [5,8]. A Narratív Kategoriális Tartalomelemző a történet kompozicionális kategóriáinak elemzésére kidolgozott automatikus eljárás. Az eljárás szemantikai,

morfológiai és szintaktikai jellemzők felhasználásával megfogalmazott szabályok alkalmazása révén a következő kompozicionális kategóriákat elemzi. A személy kategóriájában belül az én és a mi kategóriáira való utalást. A cselekvés kategóriájában annak aktivitását versus passzivitását. A szereplők belső mentális tudattartalmainak bemutatását részletesen elemzi az eljárás. Ebben a kategóriában külön azonosításra kerülnek a kognitív tudattartalmak és az érzelmi állapotok, az utóbbiak esetében a pozitív és negatív érzelmek megkülönböztetésével. Az eljárás az értékeléseket is felismeri. A pozitív és negatív minőségek ebben a kompozicionális kategóriában is megkülönböztetésre kerülnek. Az eljárás a tagadás előfordulását is azonosítja. Végül a téridői perspektíva kategóriájában az eljárás azonosítja, hogy a 3 lehetséges forma (visszatekintő, átélő és metanarratív) közül melyik érvényesül. A NarrCat elemzést kiegészítettem 2 további személy kategóriával, amelyek a termékleírást készítő személyhez közelálló (például feleség, gyermek), illetve távolálló (például szomszéd, bárki) személyek megnevezésére használatos kifejezéseket azonosítja a szövegben. A NarrCat elemzés eredménye az az adatsor, ami megadja a kompozicionális kategóriák előfordulásának számát egy-egy történetben.

4.3 Eredmények

A történetek hosszában jelentkező különbségek lehetséges hatásainak elkerülése végett a NarrCat elemzés abszolút gyakorisági adatait relatív gyakorisági mutatókká alakítottam át. Az átalakításban kétféleképpen jártam el. A téridői perspektíva kategóriában az elemzés tagmondat szinten történik, ezért ezen kategória esetében a relatív gyakoriságokat az egyes perspektívaformák előfordulásának abszolút gyakorisága és az autóleírás tagmondatokban mért számának hányadosaként számítottam ki. Az összes többi kategória esetében a NarrCat találatok abszolút gyakoriságát az autóleírás szószámban mért hosszával osztva képeztem a relatív gyakorisági mutatót.

Az elemzés első lépéseként a szövegelemzési kategóriák relatív gyakoriságát vetettük össze a Multipla és más autótípusokat értékelő szövegekben. Mivel a relatív gyakorisági mutatók eloszlása eltért a normális eloszlástól, a Mann Whitney U-próbát használtuk az összehasonlításhoz. A NarrCat kategóriák relatív gyakoriságának átlagait és standard eltéréseit, illetve különbségek tesztelésének eredményeit az 1. Táblázat tartalmazza.

Az értékelések és érzelmek előfordulására vonatkozó hipotézist alátámasztják az eredmények, mivel sem a két kategória relatív gyakoriságában, sem a pozitív és negatív értékelések és érzelmek relatív gyakoriságában nincs különbség az autóleírások két csoportja között.

A Gondolkodás kategória használatára vonatkozó hipotézist szintén alátámasztják az eredmények, mivel a Multiplák leírásaiban szignifikánsan magasabb e kategória előfordulásának relatív gyakorisága.

A személy kategóriák előfordulására vonatkozó hipotézist nagy részben alátámasztják az eredmények. A Multipla leírásokban gyakrabban jelennek meg a mi, a közeli és a távoli személy kategóriák is. Ugyanakkor az én kategória előfordulásában nincs különbség az autóleírások két csoportjában.

1. Táblázat: A NarrCat kategóriák előfordulásának relatív gyakorisága a termékleírások két csoportjában, illetve a különbségek nagyságát mutató szignifikancia értékek

NarrCat kategóriák	Multipla (N=52)		Más típusok (N=77)		Különbség
	M	SD	M	SD	Szig.
Személy (E1+T1)	,83	,34	,82	,41	,383
E1	,68	,32	,77	,39	,123
T1	,15	,19	,05	,12	**,000**
Személy (Közeli+Távoli)*	,31	,22	,14	,22	**,000**
Közeli	,17	,17	,02	,09	**,000**
Távoli	,14	,16	,11	,21	**,008**
Aktivitás	,59	,19	,54	,27	,151
Passzivitás	,10	,10	,11	,13	,355
Pszichológiai perspektíva	,89	,50	,72	,56	**,035**
Gondolkodás	,30	,23	,21	,26	**,005**
Érzelem összes	,59	,48	,52	,46	,182
Érzelem negatív	,31	,05	,27	,09	,433
Érzelem pozitív	,28	,12	,25	,12	,147
Értékelés összes	,49	,25	,49	,24	,245
Értékelés negatív	,15	,14	,14	,15	,219
Értékelés pozitív	,33	,18	,35	,21	,231
Tagadás	1,11	,45	1,23	,61	,091
Téridői perspektíva					
Visszatekintő forma	23,5	13,3	28,2	15,6	**,036**
Átélő forma	69,6	14,5	65,4	16,5	**,066**
Metanarratív forma	6,9	5,7	6,3	6,1	,296

A történetformálásra vonatkozó hipotézisünket jelentős részben alátámasztják az eredmények. A hipotézissel megegyezően a visszatekintő perspektíva forma szignifikánsan ritkábban, az átélő perspektíva forma pedig tendencia erősséggel gyakrabban jelenik meg a Multipla leírásokban, mint a más autótípusok leírásában. Ugyanakkor a metanarratív perspektíva forma használatában nincs szignifikáns eltérés az autóleírások két csoportjában.

A NarrCat kiegészített változatával végzett elemzési eljárás megbízhatóságát automatikus klasszifikáló eljárás felhasználásával értékeltem. Az automatikus klasszifikációhoz naív Bayes-féle algoritmust használtam. Mivel az elemzett autóleírások száma viszonylag alacsony volt, a keresztvalidációs eljárást választottam. Az értékelés során csak azokat a narratív kategóriákat használtam, amelyek relatív gyakorisága eltért az autóleírások két csoportjában. Ebben az esetben a találati mutató értéke 83,89 %, a pontossági mutató értéke 81,99 % volt. A két mutató értéke kis mértékben megemelkedett, amikor az összes NarrCat kategóriát, illetve a két új személy kategóriát is felhasználtam az értékeléshez. Ebben az esetben a találati mutató értéke 84,54 %-ra, a pontossági mutató értéke pedig 84,77 %-ra növekedett. Az automatikus klasszifikáció

eljárásával végzett értékelés azt mutatja, hogy a történet kompozicionális kategóriái mentén végzett elemzés képes megbízhatón megkülönböztetni a kognitív disszonancia magas és alacsony szintje mellett készített autóleírásokat.

4.4 Megbeszélés

Az elemzés eredményei azt mutatják, hogy az autóleírások szövegében azonosíthatók azok a narratív markerek, amelyek alapján megbízhatóan megállapítható az, hogy a leírás, illetve értékelés magas vagy alacsony mértékű kognitív disszonancia állapotában készült. Ez az eredmény két tanulsággal is bír a szentimentelemzés területe számára. Az egyik tanulság az, hogy nem csak az értékelést expliciten kifejező nyelvi elemek segíthetik a szentiment azonosítását, hanem az olyan elemek is, amelyek expliciten nem fejeznek ki értékelést, de fontos szerepük van a termékleírások szövegének megformálásában. Vizsgálatomban nem is jelentkezett különbség az értékelő, illetve érzelmet kifejező szavak használatának gyakoriságában, azonban a termékleírás narratív markerei közül több is összefüggést mutatott a kognitív disszonancia szintjével.

A vizsgálati eredményből levonható másik tanulság ahhoz a kérdéshez kapcsolódik, hogy az elemzett szöveg mekkora részében azonosítjuk a szentimentet. A szentimentelemzés területén elfogadott nézet szerint, ha kisebb szövegrészekben képesek vagyunk azonosítani a szentimenteket, akkor az jobb eredményre vezet [11]. Ez a stratégia minden bizonnyal növeli a szentimentelemzés pontosságát, ugyanakkor a vizsgálat eredményei azt mutatják, hogy dokumentum szinten is találhatók olyan nyelvi markerek, amelyek hatékonyan segíthetik a szövegben kifejezett szentiment megállapítását. A dokumentumszintű elemzés hatékonyságának vélhetően fontos feltétele az, hogy a szövegnek az a szerveződési szintje, ahol az elemzést elvégezzük, kapcsolódjon az értékeléshez. A történetek teljesítik ezt a feltételt, hiszen a történetek szerveződésében jelentős szerepet játszik az értékelés [7].

A vizsgálat eredményei empirikus megerősítést adnak annak az elképzelésnek is, amely azt fogalmazza meg, hogy a történet megformálásához visszatekintés kell. A kognitív disszonanciával járó erős motivációs késztetés a kognitív disszonanciát kiváltó helyzetben tartja a személyt, és így nehezíti az erre az állapotra való visszatekintést.

A vizsgálat eredményei a gyakorlati felhasználás szempontjából is jelentősek. A kognitív disszonancia jellemzően minden jelentősebb döntés, és így minden jelentősebb vásárlás után megjelenik. A magas mértékű kognitív disszonancia erős motivációs funkciója révén széleskörű hatással van a személy gondolkodására és viselkedésére. Gyakorlati szempontból ezért lehet értékes információ a termékleírást adó személy kognitív disszonanciájának szintjére következtetni, mert ez például hatással lehet a személy figyelmére, az információk feldolgozására és a személy viselkedésére is.

Az itt bemutatott elemzés azonban jelentős korlátokkal is bír. Az elemzés például csak egyetlen terméktípusra terjedt ki és abból is viszonylag kis mintát elemzett. Emellett a kognitív disszonancia mértékére vonatkozóan csak feltételezéssel tudtam élni a vizsgálatban. Az interneten hozzáférhető nagy terjedelmű termékleírások elemzése révén, illetve a személyek felkeresésével végzett pszichológiai vizsgálatok révén ezen a korlátok viszonylag egyszerűen meghaladhatók a további vizsgálatok során.

Köszönetnyilvánítás
A tanulmány az OTKA K 109009 számú pályázat támogatásával készült.

Bibliográfia

1. Aronson, E. A társas lény. Budapest: KJK-KERSZÖV Kiadó. (2001)
2. Billig, M. Ideology and Social Psychology. Oxford, Basil Blackwell. (1982)
3. Brehm, J. Post-decision changes in desirability of alternatives. Journal of Abnormal and Social Psychology. (1956) 52(3): 384–389
4. Chafe, W. Discourse, consciousness, and time. The flow and displacement of conscious experience in speaking and writing. Chicago: University of Chicago Press. (1994)
5. Ehmann B., Csertő I., Ferenczhalmy R., Fülöp É., Hargitai R., Kővágó P., Pólya T., Szalai K., Vincze O., László J. Narratív kategoriális tartalomelemzés: a NARRCAT. In Tanács A., Varga V., & Vincze V. (Szerk.), X. Magyar Számítógépes Nyelvészeti Konferencia: MSZNY 2014. Szeged: Szegedi Tudományegyetem Informatikai Tanszékcsoport. (2014) 136–147
6. Festinger, L. A Theory of Cognitive Dissonance. California: Stanford University Press. (1957)
7. Labov, W. Language in the Inner City. Studies in the Black English Vernacular. Oxford: Blackwell. (1972)
8. László, J., Csertő, I., Fülöp, É., Ferenczhalmy, R., Hargitai, R., Lendvai, P., Péley, B., Pólya, T., Szalai, K, Vincze, O., Ehmann, B. Narrative language as an expression of individual and group identity. *SAGE Open.* (2013) 3(2), 1–12
9. Pang, B., & Lee, L. Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval. (2008) 2(1–2), 1–135
10. Pólya T. Identitás az elbeszélésben. Szociális identitás és narratív perspektíva. Budapest: Új Mandátum Kiadó. (2007)
11. Szabó M. K. & Vincze V. Egy magyar nyelvű szentimentkorpusz létrehozásának tapasztalatai. In: Tanács, Attila; Varga, Viktor; Vincze, Veronika (szerk.): *XI. Magyar Számítógépes Nyelvészeti Konferencia*. Szeged: Szegedi Tudományegyetem, (2015) 219-226
12. Tedeschi, J.T.; Schlenker, B.R.; Bonoma, T.V. Cognitive dissonance: Private ratiocination or public spectacle? American Psychologist. (1971) 26(8): 685–695

Szentiment- és emóciószótárak eredményességének mérése emóció- és szentimentkorpuszokon

Drávucz Fanni[1], Szabó Martina Katalin[2,3], Vincze Veronika[2,4]

[1] Eötvös Loránd Tudományegyetem,
Bölcsészettudományi Kar, Nyelvtudományi Doktori Iskola
[2] Szegedi Tudományegyetem
szabo.martina@lit.u-szeged.hu, vinczev@inf.u-szeged.hu
[3] Precognox Informatikai Kft.
mszabo@precognox.com
[4] MTA-SZTE Mesterséges Intelligencia Kutatócsoport

Kivonat: A cikkben a szövegekben megbúvó szentiment-, valamint emotív szemantikai tartalmak összefüggéseit vizsgáljuk. A munka keretében egy kézzel annotált szentimentkorpuszt elemzünk két különböző kategóriaszámú emóciószótárral, valamint egy kézzel annotált emóciókorpuszt elemzünk egy szentimentszótár segítségével. Ezt követően a szótáras elemzésekkel kapott eredményeket összevetjük a korpuszok annotációjával. A vizsgálatok célja annak feltérképezése, hogy kiegészítheti-e, és ha igen, mennyiben a két tartalomelemzési megoldás egymás eredményeit, eredményességét. A bemutatott elemzések és eredmények egyedülállóak; nincs tudomásunk olyan dolgozatról, amely hasonló megoldásokat prezentálna. Ugyanakkor a dolgozatban amellett érvelünk, hogy a két elemzési módszer együttes vizsgálata hasznos és eddig ismeretlen eredményeket tárhat fel.

1 Bevezetés

A számítógépes nyelvészetben a szentimentek alatt a szerzői attitűdöt tükröző nyelvi elemeket [11], míg az emóciók alatt a szöveg szintjén tetten érhető érzelmeket értjük [10], melyeket a háttérben húzódó kognitív értékelő illetve emotív funkciók különböztetnek meg. E két fogalmi kategória ugyanakkor legfeljebb részben mutat átfedést. Ahogyan ugyanis arra Péter [7] rámutat, az értékelésnek létezik mind emocionális (1a), mind racionális (1b) típusa:

(1) a. a főnököm remek ember
b. a habbeton rossz hővezető

A szentimentelemzés vagy véleménykivonatolás (*sentiment analysis* vagy *opinion mining*) a természetesnyelv-feldolgozás részterülete, amely a szerzői attitűdöt tükröző nyelvi elemek detektálására, valamint értékének (*sentiment orientation*) és tárgyának (*target*) a megállapítására törekszik automatikus megoldások segítségével. Ezzel szemben az emócióelemzés (*emotion detection* vagy *emotion recognition*) a szövegekben megbúvó emóciótartalom kinyerését célozza. Jelen dolgozatban e két

tartalomelemzés feladatkörébe tartozó megoldás alkalmazásának eredményeinek az összefüggéseit vizsgáljuk, szótárak és kézzel annotált korpuszok segítségével.

Bár véleményünk szerint a két megoldás eredményei hatékonyan egészíthetik ki egymást, nincs tudomásunk olyan dolgozatról, amely e két módszert e szempontból, egymás összefüggésében vizsgálná. Ennek okát a következő sajátosságokban látjuk: Egyrészt, az érzelmek szövegalapú elemzésével csekély számú dolgozat foglalkozik a nyelvtechnológia tárgykörében (vö. pl. [8, 5]), ez összességében a magyar nyelvű szövegek elemzésére is igaznak tekinthető (vö. [2, 10, 12]). A nyelvtechnológusok kis vagy kisebb jelentőséget tulajdonítanak az emócióknak, mint az úgynevezett szentimenteknek, azaz a nyelvi értékelésnek, illetve az emóciókat a hazai nyelvtechnológia alapvetően a szentimentelemzés tárgykörébe utalja; a szentiment- és az emócióelemzés feladatát gyakran azonosítja egymással (vö. pl. [7: p202]). Ugyanakkor azt is érdemes megemlíteni, hogy a szövegek érzelmi szempontú tartalomelemzése komoly pszichológiai, nyelvészeti és nyelvtechnológiai kihívást támaszt a szakértők elé [2, 10, 12]. Figyelemre méltó azonban, hogy az érzelmek több más tudományos diszciplínában, így például a viselkedéstudományban vagy a pszichológiában központi szerepet töltenek be.

A jelen dolgozatban arra a kérdésre keressük a választ, hogy milyen összefüggés van a szövegekben levő emóciók és a nyelvi értékelés, másképpen a szentimentek között. Azt szeretnénk feltárni, hogy a két típusú szemantikai tartalom hogyan, illetve milyen mértékben mutat átfedést egymással, másképpen, mennyire jellemző a nyelvi értékelés és az emotív tartalmak összefonódása az általunk vizsgált szövegtípusokban, magyar nyelvű szövegekben.

Kutatási eredményeink [12] alapján amellett érvelünk, hogy a szövegekben megbúvó emóciótartalom kinyerése olyan értékes információkat hozhat a felszínre, amelyeket más tartalomelemző módszerek nem tárnak, illetve tárhatnak fel. Ezzel összefüggésben úgy véljük, hogy az emóció- és a szentimentelemzés módszere hatékonyabb, egymást kiegészítő tartalomelemző megoldáshoz vezethet.

2 A szótárak bemutatása

Az emóciókorpusz szentimentjeinek elemzéséhez egy saját készítésű szentiment-szótárat [9] használtunk. Szótárunkat részben automatikus, részben manuális módszerrel hoztuk létre, magyar nyelvű szövegek automatikus szótáralapú szentimentelemzése céljából . A szótár készítése során nem csupán mellékneveket, hanem főneveket, határozószókat és igéket is felvettük, amennyiben úgy ítéltük, hogy az adott nyelvi elemnek inherens negatív vagy pozitív szentimentértéke van. Az így elkészített szótárunk kutatási célokra szabadon hozzáférhető.[1]

A szövegbeni érzelmek elemzéséhez két emóciószótárat alkalmaztunk. Az egyik emóciószótárat két, kézzel készített, hat kategóriából álló szótárból (vö. [4, 10]) készítettük, a két szótár egyesítésével. A Mérő-féle gyűjtés eredetileg nem tartalmazott kategóriákat, a szótárak összefésülése érdekében azonban elemeit kategorizáltuk.

1 http://opendata.hu/dataset/hungarian-sentiment-lexicon

Mindkét szótár az emóciókifejezések osztályozásában Ekman és Friesen [1] érzelemkategorizálási rendszerét követi, tehát azt a hat alapérzelmet veszi alapul, amelyek arckifejezéseit a kutatások alapján kultúrafüggetlenül azonos módon produkáljuk és azonosítjuk. Az alapérzelmek a szerzők alapján a következők: az öröm, a düh, a bánat, a félelem, az undor és a meglepődés. A két szótár egyesítésével készült lexikon statisztikai adatait az **1. táblázat első sorá**ban közöljük.

A másik emóciólexikon, amellyel dolgoztunk, egy a hat kategóriás szótár nyolc kategóriásra bővített változata volt (vö. [12]). Az új kategóriarendszer létrehozását egy korábbi emóciókorpusz kézzel való annotációjának tapasztalata indokolta (vö. [10]). A korpusz létrehozásának célja, hogy az emóciók nyelvi viselkedését valós nyelvi anyagon vizsgálhassuk, valamint a szótáraink hatékonyságát tesztelhessük és fejleszthessük. A feldolgozói munka során azonban azt tapasztaltuk, hogy számos nyelvi elemet a meglevő hat kategóriával nem tudunk lefedni, ezért a munka második szakaszában nyolc kategóriával átdolgoztuk a meglevő teljes emóciólexikont. A két újonnan felvett kategória a feszültség és a vonzalom/szeretet volt.

A nyolc kategóriás lexikont a következő lépésekben hoztuk létre: Először mindkét, fentebb említett kiinduló szótárnak kézzel kialakítottuk a lexikáját a nyolc kategóriának megfelelően, egymástól függetlenül. A munka során az egyik kiinduló szótár (vö. [10]) anyagát – a már említett korpuszannotálási tapasztalatok alapján – további elemekkel is kiegészítettük. Ezt követően a két szótárat egyesítettük. Az így kialakított szótár statisztikai adatait az **1. táblázat második sora** mutatja be:

E.szótár	öröm	düh	bánat	félelem	undor	meglepődés	feszültség	vonzalom/szeretet	**Összesen:**
6 kat.	719	394	360	229	121	68	--	--	**1 891**
8 kat.	675	410	387	243	135	97	309	186	**2 442**

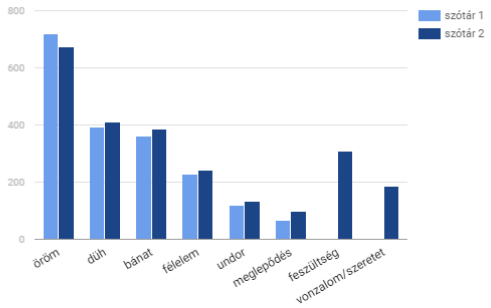
1. táblázat: Az emóciószótárak statisztikai adatai

A szótár kategóriarendszerének átszervezése az alábbi arányváltozásokat eredményezte az egyes emóciótípusokhoz tartozó elemek számában. Amint azt az 1. ábra megmutatja, egyedül az öröm kategóriába tartozó elemek száma kisebb a nyolc kategóriás szótárban, mint a hat kategóriásban. Ennek valószínűleg az az oka, hogy számos elem, amelyeket korábban az öröm-csoportban vettünk fel, megjelent az új vonzalom/szeretet kategóriában, és ezen az elemeket az öröm kategóriájából töröltük. Az összes többi emóciótípusban számbeli növekedést látni (**1. ábra**), amelynek részbeni oka az, hogy a szótár anyagát – az átkategorizáláson túl – újabb elemekkel is kiegészítettük.

3 Az elemzett korpuszok bemutatása

A kézzel annotált emóciókorpuszt kettős céllal hoztuk létre: egyrészt, hogy az emóciók nyelvi viselkedését valós nyelvi anyagon vizsgálhassuk, másrészt, hogy a

szótáraink hatékonyságát tesztelhessük és fejleszthessük [10]. Az emóciókorpusz szöveganyagát a 2014-es év folyamán keletkezett, tévés és mozis témájú blogoldalakról származó, különböző terjedelmű és szerzőségű kritikákból, hírekből, valamint kommentekből állítottuk össze, mely 15 987 mondatból és 197 707 tokenből áll. A magyar nyelvű, kézzel annotált szentimentkorpuszt termékvélemény-szövegekből hoztunk létre kutatási és fejlesztési céllal [11, 13]. Az adatbázis összesen 154 véleményszöveget, 17 059 mondatot és 251 202 tokent (központozással) tartalmaz.



1. ábra: Az emóciószótárak kategóriánkénti megoszlási arányai

4 Eredmények

Munkánk során a szentimentszótárral az emóciókorpuszt, az emóciószótárakkal pedig a szentimentkorpuszt elemeztük, azaz szótárillesztést hajtottunk végre. Amennyiben a szentimentszótárban szereplő elem illeszkedett az emóciókorpusz egyik lemmájára, akkor azt találatként értékeltük, és viszont. Az elemzések eredményeit a jelen fejezetben ismertetjük.

4.1 Az emóciókorpusz elemzése a szentimentszótárral

Az elemzéshez csupán azokat az érzelmeket vettük figyelembe, amelyek valamely konkrét emóció tagjét viselték. Amennyiben tehát egy adott elem esetében csak egy általános Emotion címke volt megadva, kihagytuk.

Az emóciókorpuszban összesen 397 annotált, emotív szemantikai tartalmú fragmentumot találtunk. Amint arról a korpusz jellemzésénél már említést tettünk (l. fentebb, 3.1), a korpuszban összesen hét emóciókategóriát annotáltunk.

Az emóciófragmentumok megoszlási arányait, valamint kategóriánként a szentimentszótárral megtalált fragmentumok számát a 2. táblázatban közöljük.

A táblázat azt mutatja meg tehát, hogy hány fragmentumot sikerült azonosítanunk a pozitív és a negatív szentimentszótárunk segítségével. A százalékos adatok azt jelzik, az összes fragmentumból hányat talált meg minimum egy elem a negatív, és hányat a

pozitív szótárból. A kapott adatok alapján a következő megállapításokat tehetjük: A szentimentszótárral az annotált emóciókifejezések megtalálási aránya jelentősnek tekinthető. A negatív szólistával az elemek 52,4%-át, a pozitív listával 41,1%-át sikerült detektálni. A legmagasabb eredményeket a negatív lexikonnal a bánat (72,3%) és a feszültség (71,2%) esetében, míg a pozitív lexikonnal az öröm (73,2%) esetében értük el.

Emóciókorpusz		**Szentimentszótárral azonosított elemek száma**	
tag	**fragmentumok**	**negatív**	**pozitív**
harag	58	35: 60,3%	22: 37,9%
feszültség	73	52: 71,2%	15: 20,5%
undor	11	8: 72,7%	5: 45,5%
félelem	25	16: 64,0%	10: 40,0%
öröm	112	25: 22,3%	82: 73,2%
bánat	83	60: 72,3%	22: 26,5%
meglepetés	34	12: 35,3%	7: 20,6%
Összesen:	**397**	**208: 52,4%**	**163: 41,1%**

2. táblázat: Az emóciókorpuszon végzett szentimentszótáras elemzés eredményei

Annak tekintetében, hogy a különböző emóciókat mely szótárral találtuk meg, és ezek között a találatok között milyen a megoszlási arány, a következőket mondhatjuk el: A találati arány a legtöbb esetben tükrözi az adott érzelem polaritását, azaz pozitív vagy negatív voltát. A meglepetés találati kiegyenlítettsége érthető, tekintettel arra, hogy ez az egyetlen alapemóció, amely pozitív és negatív egyaránt lehet. Ugyanakkor az figyelemre méltó, hogy a harag, az undor és a félelem emóciófragmentumokat is nagy arányban detektálja a velük ellentétes polaritású szentimentlexikon.

A **3. táblázat** megmutatja az eredményeket úgy, ha az emóciókategóriákat polaritásuk alapján összevonjuk. A meglepetés kategóriát, a fentebb említett ok miatt külön sorként tüntetjük fel.

Emóciókorpusz		**Sz.szótárral azonosított elemek száma**	
tag	**fragmentumok**	**negatív**	**pozitív**
negatív	250	171: 68,4%	74: 29,6%
pozitív	112	25: 22,3%	82: 73,2%
meglepetés	34	12: 35,3%	7: 20,6%

3. táblázat: Az emóciókorpuszon elemzése összevont szentimentszótárral

Úgy véljük, hogy a jelenség oka – legalább részben – a negáló elemek használatában keresendő: valószínűleg gyakran fejezzük ki negatív érzelmeinket pozitív polaritású elemek tagadásával, és ezekben az esetekben a fragmentum polaritása nem egyezik meg a benne szereplő emóciókifejezés polaritásával

(pl. *egyáltalán nem örülök, nem volt elragadtatva*). A tapasztalatokat alaposabb vizsgálat tárgyává kívánjuk tenni a jövőben.

4.2 A szentimentkorpusz elemzése az emóciószótárral

4.2.1 Illesztés

A szótáralapú elemzés során a hat, majd a nyolc kategóriás emóciószótárat is a szentimentkorpuszra illesztettük. A szentimentkorpusz 15 675 fragmentumából a hat és a nyolc emóciót tartalmazó szótár alapján összesen 4 310 és 4 380 fragmentumot találtunk meg. Egy fragmentumon belül bizonyos esetekben több illeszkedése is volt a szótárnak. A hat kategóriás szótár esetében összesen 4586 illeszkedése volt a 4 310 fragmentumnak, illetve a nyolc kategóriás szótárnál 4 682 illeszkedés 4 380 fragmentumban. Részletesen lásd a **4-5. táblázatok**at.

Szentimentkorpusz		**Emóciószótárral azonosított fragmentumok**						
tag	**fragmen-tumok**	**bánat**	**düh**	**félelem**	**undor**	**megle-petés**	**öröm**	**Össz.:**
negatív	8 465	386; 4.6%	54; 0.6%	225; 2.7%	556; 6.6%	40; 0.5%	632; 7.5%	1 700; 20,1%
pozitív	7 210	86; 1.2%	18; 0.2%	37; 0.5%	392; 5.4%	97; 1.3%	2 063; 28.6%	2 610; 36,2%
Össz.:	**15 675**	**472; 3.0%**	**72; 0.5%**	**262; 1.7%**	**948; 6.0%**	**137; 0.9%**	**2 695; 17.2%**	**4 310; 27.5%**

4. táblázat: A hat kategóriás emóciószótár illesztése a szentimentkorpuszon

Sz.korpusz		**Emóciószótárral azonosított fragmentumok**								
tag	**frag-men-tumok**	**bánat**	**düh**	**félelem**	**feszült-ség**	**undor**	**meg-lepetés**	**öröm**	**sze-retet**	**Össz:**
negatív	8 465	386; 4,6%	41; 0,5%	225; 2,7%	52; 0,6%	556; 6,6%	40; 0,5%	545; 6,4%	101; 1,2%	1 739; 20,5%
pozitív	7 210	86; 1,2%	13; 0,2%	36; 0,5%	34; 0,5%	392; 5,4%	97; 1,3%	1 786; 24,8%	292; 4,0%	2 641; 36,6%
Össz:	**15 675**	**472; 3,0%**	**54; 0,3%**	**261; 1,7%**	**86; 0,5%**	**948; 6,0%**	**137; 0,9%**	**2 331; 14,9%**	**393; 2,5%**	**4 380; 27,9%**

5. táblázat: A nyolc kategóriás emóciószótár illesztése a szentimentkorpuszon

A nyolc kategóriás emóciószótár, bár 29,1%-kal nagyobb a hat kategóriásnál, csupán 70-nel (0,4%) több fragmentum annotációját eredményezte.

A szentimentkorpuszbeli fragmentumoknak körülbelül a negyedében (27,9% illetve 27,5%) azonosítottunk a két szótár segítségével emóciókifejezést. A korpusz szövegtípusaira tekintettel, amely termékvéleményeket és twitter bejegyzéseket, tweeteket tartalmazott megállapítható, hogy az emóció nem tipikus, illetve domináns formája a

szentiment kifejezésének, tehát a szentimentkorpusz a fenti eredmények alapján többnyire tárgyilagosnak tekinthető. A kapott adatokra támaszkodva a jövőben az objektivitás kérdését tovább kívánjuk vizsgálni más tartalomelemzési megoldásokkal (pl. funkciószó-megoszlás) is hasonló domainen.

A pozitív szentimentet tartalmazó fragmentumokban 16,1%-kal volt magasabb az emóciókifejezések aránya, ezen belül nem meglepő módon a legtöbb illeszkedést a pozitív emóciók (öröm, szeretet) eredményezték. Az aránytalanság lehetséges magyarázata, hogy a pozitív érzelmek kifejezése kisebb lexikai változatosságot mutat, ezért a szótárral való illesztés a szűkebb nyelvi eszközkészletet nagyobb arányban képes azonosítani. Ezt alátámasztja az emóciószótárban felülreprezentált negatív emóciók száma is. Lehetséges magyarázat még a politikai korrektség jelensége, azaz hogy az adatközlők a negatív véleményt a nyilvánosság miatt árnyaltabban, indirekt módon fejezik ki – vagy akár vissza is tartják. Ezenkívül feltételezhető, hogy mivel a korpusz alapjául szolgáló szövegek szerzői a felületet online közösségi felületként használják, ezért a közösségből való kizárás elkerülése végett tartózkodnak a potenciálisan megosztó, azaz szélsőséges, direkt érzelemkifejezéstől. E feltételezés alátámasztására más jellegű (nem online) közlésekből összeállított korpusz összehasonlító elemzése szükséges, amely feltárhatja, hogy a jelenség milyen mértékben jellemző a korpuszra, vagy általában a magyar nyelvű, írott formában megjelenő érzelemkifejezésre.

Minden egyes emóciónak mind a két szentiment fragmentumaiban volt illeszkedése, a legtöbb illeszkedése az öröm emóció kifejezésének, míg legkevesebb a düh emóció kifejezésénél figyelhető meg.

Tetten érhető a korreláció az illeszkedések emóciójának polaritása és az illeszkedő szentiment fragmentum polaritása között, azaz például a bánat, illetve az öröm ~4-szer nagyobb arányban illeszkedik a negatív, mint pozitív szentimentű fragmentumokra. A meglepődés az elemzett korpuszban többségében pozitív polaritású, tehát nem várt jó dologra vagy tulajdonságra utalt, mivel 2,5-szer nagyobb arányban illeszkedett a pozitív szentimentekre. Az undorhoz tartozó illeszkedéseknek csak 58,6%-a volt negatív, amelynek hátterében a negáció, vagy ellentételezés alkalmazása is lehet, ennek elemzését lásd később.

A következő szakaszban megvizsgáljuk, hogy ezen korrelációkat kihasználva a szentimentek polaritását az emóciószótárral mennyire pontosan lehet előrejelezni.

4.2.2 Előrejelzés illesztés alapján

Az előző szakaszban az illeszkedő emóciókifejezések és a szentiment fragmentum tag polaritása közötti korreláció alapján arra adódik lehetőség, hogy előrejelezzük a szentimentet az illeszkedő emóció fragmentumok segítségével.

Emóciónként az egyes mondatokra akkor jeleztünk előre az adott emócióval azonos polaritású szentiment taget, ha volt illeszkedés az adott emócióhoz tartozó fragmentumokkal. A meglepetés polaritása – ahogy arra már korábban is utaltunk (l. fentebb, 4.2.1) – nem egyértelmű, mégis a fentebb bemutatott vizsgálat során azt tapasztaltuk, hogy a jelen korpuszban pozitív polaritású öröm emócióhoz hasonlóan felülreprezentált a pozitív szentimentű mondatok között, így a meglepetés emóciókifejezés illeszkedése esetén pozitív szentimentet jeleztünk előre.

Az emóciónkénti előrejelzés mellett kombinált előrejelzést is végeztünk, ahol bármely negatív emóciókifejezés illeszkedése esetén a negatív szentiment taget jeleztünk

előre. Az emóciónkénti, valamint a kombinált módszerrel kapott előrejelzési eredményeket lásd a **6-7. táblázatok**ban.

A várakozásoknak megfelelően a pontosság lényegesen magasabb a fedésnél, hiszen az emócióilleszkedés a szentiment-fragmentumok kevesebb mint 30%-ban fordult elő (l. fentebb, 4.2.1). A nyolc kategóriás emóciószótár kombinált pontossága alacsonyabb a hat kategóriás szótárénál, míg fedése a kombinált esetében magasabb, aminek hátterében a nagyobb szótárméret valószínűsíthető.

Emóciószótár	Szentiment	Pontosság (P)	Fedés (F)	F1
bánat	negatív	81,8%	4,6%	8,6%
düh	negatív	75,0%	0,6%	1,3%
félelem	negatív	85,9%	2,7%	5,2%
undor	negatív	58,6%	6,6%	11,8%
meglepetés	pozitív*	70,8%	1,3%	2,6%
öröm	pozitív	76,5%	28,6%	41,7%
Kombinált	**negatív**	**67,3%**	**12,8%**	**21,5%**
	pozitív	**76.0%**	**29.4%**	**42.4%**

6. táblázat: A hat kategóriás emóciószótár osztályozási eredményei

A nyolc kategóriás szótár pontossága az egyes emóciók esetén mindössze 2 százalékponton belüli eltérést mutatott a hat kategóriás szótárral való illesztéshez képest. A legnagyobb pontosságot mindkét szótár esetén a félelem emóció mutatta (85,6% illetve 86,2%), míg a legalacsonyabb pontosságot az undor (58,6%).

A hibaelemzéssel megállapítottuk, hogy az undor emóció 391 pozitív szentimentű fragmentumra illeszkedett (fals pozitív hiba), amelyből 358 (92%) fragmentum tartalmazott negációt (például: „Az íze nem rossz”, „Ezzel a szaloncukorral biztosan nem fog rosszul járni”). Ez alapján valószínűsítettük, hogy a negáció figyelembevételével tovább javítható az előrejelzés pontossága.

A hat és nyolc kategóriás szótár esetében is megfigyelhető, hogy míg a pozitív szentimentű fragmentumok több mint negyedében illeszkedett vele azonos polaritású emóciókifejezés, addig a negatív szentimentet tartalmazó fragmentumok esetében csak a nyolcada mutatott azonos polaritású illeszkedést. Ez arra enged következtetni, hogy a korpuszban diverzebbek, konfúzabbak a negatív érzelmeket kifejező nyelvi szerkezetek.

A hat kategóriás emóciószótárban az undor érzelmet kifejező emóciófragmentumok negatív szentimentek előrejelzésének a pontossága önmagában a legalacsonyabb (58,6%). Ha figyelembe vesszük a tagadószavakat, ez az érték lényegesen javul (93,9%). További kutatási célként megfogalmazható, hogy érdemes megvizsgálni az undort kifejező emóciófragmentum azon eseteit, amikor negáció kapcsolódik hozzá.

A meglepetés emóciószótár a szentiment korpuszon jobban jelezte előre a pozitív szentimentet, mint a negatívat annak ellenére, hogy az emóciót semlegesnek gondoljuk. A meglepetés emóciószótár alapvetően semlegesnek tekinthető (pl. *nem számít rá, hihetetlen*), de van néhány polarizált kifejezés (pl. *csodál, megilletődés, szörnyülködés, lefagy*) is.

Emóciószótár	Szentiment	Pontosság (P)	Fedés (F)	F1
bánat	negatív	81,8%	4,6%	8,6%
düh	negatív	75,9%	0,5%	1,0%
félelem	negatív	86,2%	2,7%	5,2%
feszültség	negatív	60,5%	0,6%	1,2%
undor	negatív	58,6%	6,6%	11,8%
meglepetés	pozitív*	70,8%	1,3%	2,6%
öröm	pozitív	76,6%	24,8%	37,4%
szeretet/vonzalom	pozitív	74,3%	4,0%	7,7%
Kombinált	**negatív**	**66,9%**	**13,2%**	**22,1%**
	pozitív	**75,7%**	**29,4%**	**42,4%**

7. táblázat: A nyolc kategóriás emóciószótár osztályozási eredményei

A hat és nyolc kategóriás emóciószótárak szentiment előrejelzése hasonló eredményeket mutatott. Mind a nyolc, mind a hat kategóriás szótárban voltak olyan fragmentumok, amelyekben az emóció több szótárilleszkedést is mutatott. A nyolc kategóriás szótárnál 98 olyan fragmentum volt, amely több emóció esetén is mutatott szótárbeli illeszkedést, (pl: „*Borzalmas* gagyi csoki darabok vannak benne", „*Csodá-san* kipárnázott nagyon jó futócipő") de ezek túlnyomórészt azonos szentiment-polaritású mondatban fordultak elő, azaz jól osztályoztak. Ezek az elemek (a példákban eltérő szedéssel emeltük ki) a különböző szótáralapú tartalomelemzési feladatokban azért problémásak, mert a lexikai szintű emotív tartalmuknak, illetve polaritásuknak megfelelően szerepelnek az emóció- vagy a szentimentszótárban, és szótáras elemzésük is ennek megfelelően történik. A problémáról l. [14].

Kivételként hozható fel például a negatív szentiment fragmentumok és az öröm emóció illesztése („mégsem nyerte el a *tetszésünket*", „Az amúgy nagyon jó reklámai miatt már-már jó sörnek tűnő [márkanév] a vakteszten jó nagyot bukott") vagy a pozitív szentiment-fragmentum és undor emóció illesztése („*rossznak* sem *rossz*").

A magas pontosság azt sejteti, hogy az emóció és a szentiment egymástól nem független, viszonyuk úgy írható le, hogy az emócióból következik a szentiment, azaz az emóciók az egyes szentimentek alfajai.

A tapasztalatok alapján az előrejelzéseket megismételtük úgy, hogy az emóciószótár-beli illeszkedés és negáció együttes jelenléte esetén is az adott emócióval ellentétes polaritású szentimentet jeleztük előre, azaz negáció jelenléte esetén figyelmen kívül hagytuk az illeszkedést. Például az „íze nem *rossz*" fragmentum esetén bár illeszkedik a „rossz" kifejezés ami az undor emóciószótárban megtalálható, a negáció miatt mégis az undorral ellentétes, tehát pozitív szentimentet jelzünk előre. Az így kapott eredményeket lásd a **8-9. táblázatok**ban.

A várakozásoknak megfelelően a negáció figyelembevétele még az együttes előfordulás naiv megközelítésével is átlagosan 14,9, illetve 13,2 százalékponttal növelte az emóciónkénti pontosságot, a fedés törvényszerű csökkentése mellett.

A tagadószavak figyelembevétele nem javította ugyanakkor a meglepetés emóció illeszkedésének pontosságát, ahol enyhe csökkenés volt (kevesebb, mint 1%). Ezt

részben magyarázhatja, hogy az emóciószótárban vannak olyan kifejezések, amelyek negációt tartalmaznak. Ezt a jelen (naiv) módszer nem kezelte megfelelően.

Emóciószótár	Szentiment	Pontosság (P)	Fedés (F)	F1
bánat	negatív	89,8%	3,5%	6,8%
düh	negatív	93,9%	0,5%	1,1%
félelem	negatív	90,0%	2,4%	4,7%
undor	negatív	93,9%	6,0%	11,3%
meglepetés	pozitív*	70,1%	1,2%	2,4%
öröm	pozitív	88,7%	27,1%	41,5%
Kombinált	**negatív**	**91,1%**	**11,0%**	**19,7%**
	pozitív	**87,6%**	**27,8%**	**42,2%**

8. táblázat: A hat kategóriás emóciószótár eredményei negáció kezelésével

Emóciószótár	Szentiment	Pontosság (P)	Fedés (F)	F1
bánat	negatív	89,8%	3,5%	6,8%
düh	negatív	92,5%	0,4%	0,9%
félelem	negatív	90,0%	2,4%	4,7%
feszültség	negatív	90,5%	0,4%	0,9%
undor	negatív	93,9%	6,0%	11,3%
meglepetés	pozitív*	70,1%	1,2%	2,4%
öröm	pozitív	88,7%	23,5%	37,2%
szeretet	pozitív	88,7%	3,8%	7,3%
Kombinált	**negatív**	**91,0%**	**11,3%**	**20,1%**
	pozitív	**87,6%**	**27,8%**	**42,2%**

9. táblázat: A nyolc kategóriás emóciószótár eredményei negáció kezelésével

A negációra történő szűrés után körülbelül egyforma volt a pontossága a hat kategóriás és nyolc kategóriás szótárnak, természetesen az előfordulási gyakoriság minden emóció esetében másképp alakul. A meglepetés ennél az esetnél is kivételt képez, az *elképesztő(en), hihetetlen(ül)* intenzifikálói funkcióban való használata miatt, amikor is a vizsgált elemek nem a lexikai szintű polaritásukat hordozzák, hanem puszta fokozó szerepet töltenek be (vö. [14]).

4.2.3 Részletes hibaanalízis

A szentimentkorpusz emóciószótárakkal történő vizsgálata során az előrejelzési hibákat és kapcsolódó észrevételeket a következő kategóriákba soroltuk. *Szótári hibák*nak neveztük azokat az eltéréseket, amikor az illesztés a szótár tartalmára visszavezethetően elmaradt, vagy nem a megfelelő emócióra vonatkozott:

- Hiányos szótár vagy téves illesztés állandósult szókapcsolatok, szólások esetén: „A többi olyan , mint halottnak a *csók*”, „Te *jó* ég”, „wow”,)
- Azonos alakú („Folyékony *zsír*”, „*csípi* a torkom”)

- Többtagú kifejezések („kellemes *csalódás*”, „nem *rossz*_”)
- Félrekategorizálás („Jaj” - a bánatban van, de meglepődésnél nincsen)
- Eltérő szófaj („feldobottság” vs. „feldob”; „szeretet” vs. „szeret”)
- Hangulatfestő („fúúúúj”, „ööö”)
- *Módszertani hibák*nak neveztük azokat a jelenségeket, amikor a szótár megfelelő volt, de az illesztésnél eltérés volt tapasztalható („*Tökéletesen* közepes”, „csak *fél* ponttal csúszott le a harmadik helyről”; nincs, de kellene „jól esett”, „meg vagyok elégedve”, „bejön”, „fúj”).

A következő csoportba tartoznak azok a hibák, amelyek különböző nyelvi jelenségek nem megfelelő kezeléséből adódtak:

- Óhajtó értelem (pl. „lehetne sokkal *jobb* is”)
- Szarkazmus (pl. „ha használhatatlan zsebkendőre vágyik , ez lesz az ön *ideális* ár / érték arányú terméke”)
- Kulturálisan kódolt jelenségek („műanyag íz”, „*kipirosítva* az orrot”)
- Kontextus ismeretére lenne szükség a pontos meghatározáshoz („Ebben *tuti*, hogy 20%-os ecet van”)
- Ellentétes polaritású határozó (pl. „*iszonyú* trendi”, „*borzasztó* finom amellett”, „*Elképesztően* kellemetlen íze van”, ”fázott is *rendesen* a lábam„)
- Burkolt negáció, viszonyítás („Voltak *jobbak*”, „Több rosszat kapott, mint *jót*”, „feltételeztem, a hús is *jó* benne”, „pont annyira *keserű*, amennyire kell”)
- „Jó”, „szépen” mint negatív-nyomatékosító (pl. „Sűrű a *jó* sok zselatintól”, „az a mennyiség most *szépen* ki is engedett”)
- Kettős tagadás („túl bizarr , hogy *utálni* lehessen”)

5 Összegzés

Munkánkban a szövegekben megbúvó szentiment-, valamint emotív szemantikai tartalmak összefüggéseit vizsgáltuk, melynek során kézzel annotált szentiment- illetve emóciókorpuszt elemzünk különböző emóció- illetve szentimentszótárakkal. A szótáras elemzésekkel kapott eredményeket összevetése a korpuszok annotációjával rejtett összefüggésekre mutatott rá. Például az egyszavas emóciószótár-illeszkedések a szentimentkorpuszon a fragmentumok alig negyedében fordultak elő, de már pusztán a negáció együttes előfordulásának figyelembevételével is 90% körüli pontosságot mutattak a szentiment polaritásának azonosításában. A manuális hibaanalízis több módszertani, nyelvi, pragmatikai és kognitív okot tárt fel, melyek magyarázzák a pontosságot csökkentő fals illeszkedéseket és lehetővé teszik a bemutatott és hasonló módszerek továbbfejlesztését. Ugyanakkor az elemzés során nem került azonosításra olyan fragmentum, amely a kontextus ismerete nélkül is egyértelműen a jelölt szentimenttel ellentétes polaritású érzelmet jelenített volna meg, így jelen korpusz esetében az emóciók a szentimentek aleseteinek tekinthetők a fragmentumok szintjén.

A bemutatott elemzések és eredmények egyedülállóak; nincs tudomásunk olyan hazai munkáról, amely hasonló megközelítést vizsgálna. A vizsgálat során feltérképeztünk olyan kutatási, alkalmazási és továbbfejlesztési lehetőségeket, amelyekben kiegészítheti egymást a két tartalomelemzési megoldás.

Köszönetnyilvánítás

Jelen kutatás az Emberi Erőforrások Minisztériuma (EMMI) Új Nemzeti Kiválóság Program (ÚNKP) támogatásával valósult meg.

Bibliográfia

1. Ekman, P., Friesen, W.V. 1969. The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica* 1: 49–98.
2. László J., Ehmann B. 2004. A narratív pszichológiai tartalomelemzés új eljárása: A LAS-Vertikum. In: Erős F. (szerk.): *Magyar Pszichológiai Szemle Könyvtár: Az elbeszélés az élmények kulturális és klinikai elemzésében.* Akadémiai Kiadó, Budapest. 75−87.
3. Liu, B. 2012: *Sentiment Analysis and Opinion Mining.* Draft. Elérhető: http://www.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf
4. Mérő L. 2010. *Az érzelmek logikája*. Tericum, Budapest
5. Mulcrone, K. 2012. Detecting Emotion in Text. Elhangzott: UMM CSci Senior Seminar Conference. Amerikai Egyesült Államok, University of Minnesota: Morris. 2012. ápr. 28. https://wiki.umn.edu/pub/UmmCSciSeniorSeminar/Spring2012Talks/KaitlynM ulcrone.pdf
6. Péter M. 1991. *A nyelvi érzelemkifejezés eszközei és módjai.* Tankönyvkiadó, Budapest.
7. Pólya T., Csertő I., Fülöp É., Kővágó P., Miháltz M., Váradi T. 2015. A véleményváltozás azonosítása politikai témájú közösségi médiában megjelenő szövegekben. In: *XI. Magyar Számítógépes Nyelvészeti Konferencia*. 198–209.
8. Strapparava, C., Mihalcea, R. 2008. *Learning to identify emotions in text.* SAC 2008. http://web.eecs.umich.edu/~mihalcea/papers/strapparava.acm08.pdf
9. Szabó M.K. 2015. Egy magyar nyelvű szentimentlexikon létrehozásának tapasztalatai és dilemmái. In: *Nyelv, kultúra, társadalom. Segédkönyvek a nyelvészet tanulmányozásához* 177. Tinta, Budapest. 278–285.
10. Szabó M.K., Morvay G. 2015. Emócióelemzés magyar nyelvű szövegeken. In: *Nyelv, kultúra, társadalom. Segédkönyvek a nyelvészet tanulmányozásához* 177. Budapest, Tinta. pp. 286–292.
11. Szabó M. K., Vincze V. 2015. Egy magyar nyelvű szentimentkorpusz létrehozásának tapasztalatai, In: *XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015)*. Szegedi Tudományegyetem, Szeged. 219−226.
12. Szabó M.K., Vincze V., Morvay G. 2016a. Magyar nyelvű szövegek emócióelemzésének elméleti nyelvészeti és nyelvtechnológiai problémái. In: *Távlatok a mai magyar alkalmazott nyelvészetben.* Budapest: Tinta
13. Szabó M.K., Vincze V., Simkó K., Varga V., Hangya V. 2016b. A Hungarian Sentiment Corpus Manually Annotated at Aspect Level. In: *Proceedings of LREC 2016.* Portoroz, Szlovénia Portoroz: European Language Resources Association (ELRA). 2873-2878.
14. Szabó M.K. *The usage of elements with emotive semantic content from a gender point of view.* Kézirat.

Entitásorientált véleménykinyerés magyar nyelven

Huszti Dániel és Ács Judit

Budapesti Műszaki és Gazdaságtudományi Egyetem,
Automatizálási és Alkalmazott Informatikai Tanszék,
huszti.daniel@gmail.com, judit@aut.bme.hu

Kivonat Napjainkban a digitális formában fellelhető, strukturálatlan adatok mennyisége folyamatosan növekszik, ezáltal a bennük említett entitásokra vonatkozó vélemények polaritásának automatizált elemzése is egyre fontosabbá válik. Cikkünkben bemutatunk egy olyan alkalmazást, mely segítségével magyar nyelvű szövegekből lehetséges a tulajdon-, földrajzi- és cégnevekre vonatkozó, részletes szerzői attitűd kinyerése. A forráskódot és a megoldást virtualizált formában is nyilvánosságra hoztuk.
Kulcsszavak: véleménykinyerés, polaritás, szentiment, természetesnyelv-feldolgozás

1. Bevezetés

Az információs társadalom által generált szöveges adatok mennyiségének drasztikus növekedésének köszönhetően az automatizált elemzési megoldások egyre szélesebb körben kezdtek elterjedni. Ezen igaz a véleménykinyerés területére is, a szövegrészletekben előforduló különböző entitásokra (tulajdonnevek, földrajzi és cégnevek) lebontva, részletes kimutatások előállítására mutatkozik jelentős piaci igény.

Magyar nyelvre publikusan elérhető szentiment korpuszok száma csekély, ezek közül entitás-szintű véleménykinyerésre egyedül az OpinHuBank alkalmas. Munkánk során ez utóbbi korpuszt felhasználva úgy tanítottuk be a modellünket, hogy képes legyen az egyes entitásokra vonatkozó polaritások megállapítására. Az implementáció során törekedtünk a valós életben történő alkalmazhatóságra, ezért az iteratív fejlesztési folyamat során valós példákon is megvizsgáltuk az éppen aktuális modell pontosságát. A nyilvánosságra hozott alkalmazásban moduláris felépítést alkalmaztunk a továbbfejleszthetőség érdekében. Az intuitív módon használható fejlesztői interfész és a Docker container technológia által garantált platformfüggetlen futtathatóság nagyban segítheti az applikáció felhasználását.

2. Létező megvalósítások

A természetes nyelvfeldolgozás, azon belül a véleménykinyerés napjaink egyik legnépszerűbb kutatási területévé emelkedett, melyet a nemzetközi versenyekre és konferenciákra benyújtott számos koncepción felül a nagyvállalati megoldások jelenléte is alátámaszt. Utóbbira jó példa a világ egyik legnagyobb videostreaming szolgáltatója, mely valós időben történő véleménydetektáló rendszer integrálásával biztosít interaktív videózási élményt.

Az egyik legjelentősebb megmérettetés az Association for Computational Linguistics (röviden ACL) intézet által szervezett SemEval [1] [2] [3], amely

évről-évre egyre több jelentkezőt vonz, akik több különféle komplexitású feladatban is összemérhetik megoldásaik hatékonyságát. Az utóbbi három évben egyre nagyobb jelentőséget kapott a véleménykinyerés szekció, azon belül pedig az aspektus-szintű elemzés, eleinte csak mondatszintű, majd szövegrészletre kiterjesztett, 2016-ban pedig már domainen túlívelő szentiment analízis feladatok is kitűzésre kerültek.

A mondatszintű véleménydetekciós megmérettetés alapja az elmúlt három évben változatlan, az éttermekre és laptopokra vonatkozó értékelésekből az aspektusokhoz (pl. étel vagy kiszolgálás minősége) tartozó vélemények kategóriájának definiálása a cél. Megvizsgáltuk az ott részletezett koncepciókat, a legjobbak háromosztályos aspektus-szintű szentiment elemzésre 80% feletti pontosságot tudtak elérni. Ezen megoldások alapkoncepciója majdnem minden esetben azonos, az alapvető nyelvi elemzés eszközei segítségével mondat- és szóhatárok, szófaj és morfológiai felbontás meghatározását, majd a funkciószavak kiszűrését követően a szótövezett alakokra unigram, néhol bigram jellemzők illetve feladatspecifikus súlyozás vagy dimenziócsökkentés kerül alkalmazásra. Utóbbi kettő a polaritás szempontjából érdekes szavak kiemelésére használatos technika. Általánosságban vett optimális megoldás nem létezik, mivel gyakran egyedi, a korpuszra jellemző tulajdonságokat vesznek figyelembe.

Megvizsgáltunk egy cseh nyelvre elkészített megoldást is [4], melyben uni- és bigram jellemzők szótövezett és az eredeti alakját jegyként felhasználva 66,27%-os pontosságot értek el háromosztályos aspektus-szintű szentiment elemzésre.

A korszellemnek megfelelően számos neurális hálós megoldás is született entitásorientált véleménykinyerésre[5,6].

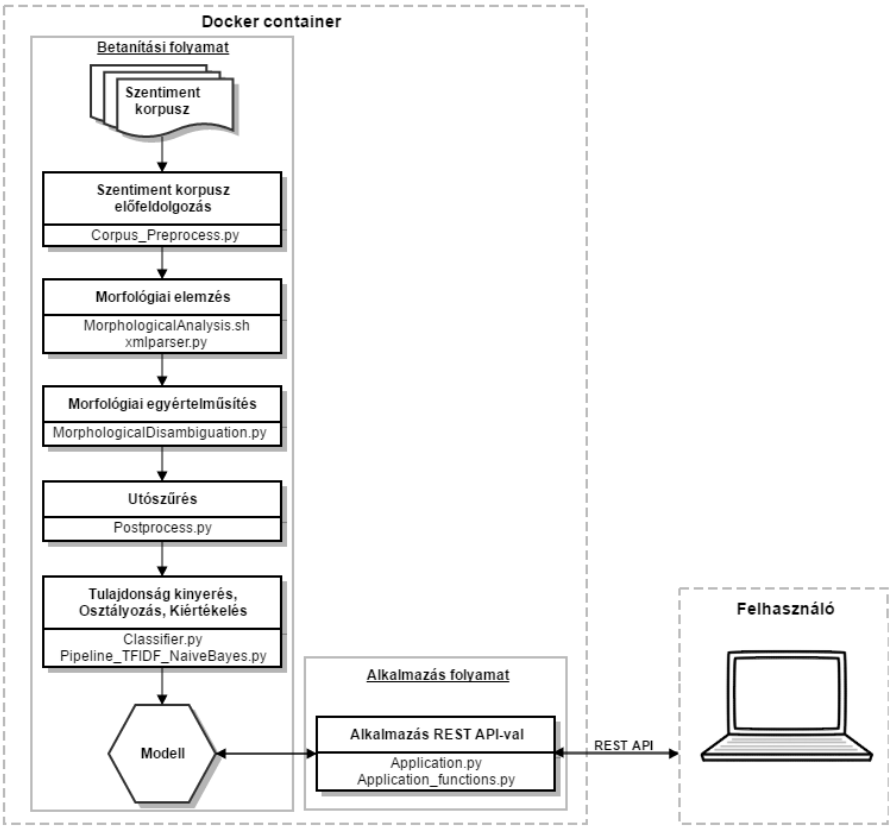
Magyar nyelven talán a *Trendminer* [7,8] a legismertebb megoldás, amely az OpinHuBank szentiment korpuszon uni- és bigram jellemzők felhasználásán felül speciális, távolság alapú súlyozás illetve polaritáslexikonok segítségével három- és kétosztályos esetben is 80% feletti pontosságot ér el.

Az imént említett magyar nyelvre implementált megoldások forráskódját nem hozták nyilvánosságra, ezért úgy gondoltuk, hogy érdemes egy a SemEval aspektus-szintű véleménykinyerés feladatához, és a [9] cikkhez hasonló, nyílt forráskódú alkalmazást elkészíteni, mely képes a szabad entitásokhoz kapcsolódó vélemények kategorizálására. Mivel a magyar nyelvre elkészített, publikusan elérhető szentiment korpuszok száma nagyon korlátozott, ezért az egyetlen, ilyen entitás-mondat párokat tartalmazóra, az OpinHuBank adatbázisra [10] esett a választásunk. A főként internetes hírportálokról, blogokról letöltött mondatokat öt természetes személy annotálta pozitív/semleges/negatív kategóriák egyikébe. Az entitás jelen esetben mindenképpen egy természetes személy, azonban ez attól még jó alapként szolgálhat a modell későbbi általánosítása céljára, így az kis módosítással akár termékekről keletkező vélemények elemzésére is alkalmassá válhat.

Tudomásunk szerint az egyetlen szabadon elérhető magyar nyelvű véleménykinyerő a Polyglot sentiment analysis modulja [11], ami támogatja a magyar nyelvet is, amellyel össze is hasonlítottuk a mi megoldásunkat.

3. Alkalmazott módszerek

Az alkalmazásunk felépítését a 1 ábra szemlélteti. Alapvetően három részre bontható: egy előfeldolgozó, egy nyelvfeldolgozó vagy NLP és egy gépi tanuló modulra.

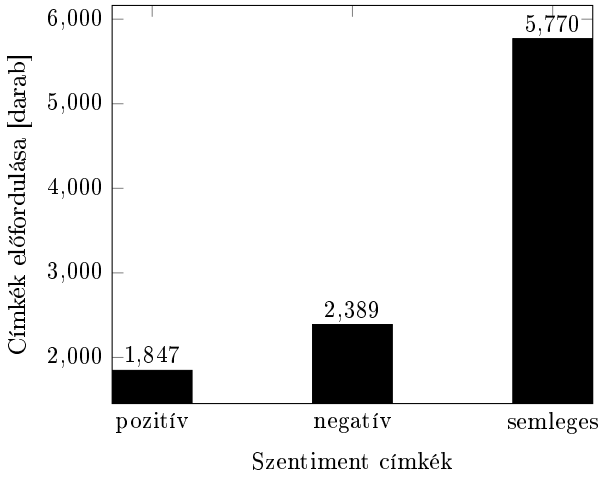


1. ábra. Alkalmazás felépítése

A nyelvfeldolgozó modul megvalósításához a BME MOKK Hun* eszközeit, a PrecoSenti magyar szentiment lexikonjait[1], a Polyglot NER tulajdonnév detektálásra szolgáló eszközét [12] alkalmaztuk, míg a gépi tanuló modulhoz a Python Sklearn [13] csomagját használtuk fel. A betanítás előtt a tanító és teszt adathalmaz 80-20% arányban történő véletlenszerű szétválasztását alkalmaztuk. A tulajdonságkinyerést és a tanító algoritmus futtatását az Sklearn Pipeline segítségével automatizálva végeztük. Az optimális paraméterek kiválasztását hasonlóképp az Sklearn GridSearchCV funkciója segítségével, tízszeres keresztvalidációval határoztuk meg.

3.1. Korpusz előkészítése

Az OpinHubank 5 annotátor értékeléseit tartalmazza, akik közt az egyetértés nagyon változó. Az annotátorok által adott pontszámokat összeadtuk, azonban így is az entitások 57,76%-a kapott semleges értékelést, míg negatív (-5– -1), illetve pozitív (1 – 5) értékelést egy pontszámra levetítve nagyon kevés entitás kap (200–500 kategóriánként). A negatív és pozitív pontszámokat egyetlen negatív, illetve pozitív kategóriára vetítettük le, ezáltal három osztályt hoztunk létre (negatív, semleges, pozitív). Az osztályok eloszlását a 2 ábra szemlélteti.



2. ábra. OpinHuBank mondatainak kategorizálása

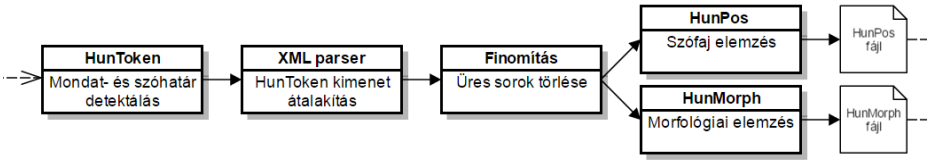
Az optimális megoldás megtalálása érdekében elvégeztem a három – pozitív/semleges/negatív – kategóriára történő szűkítést, azaz az előző kalkulált érték alapján nullánál nagyobb értékkel szereplő előfordulásokat pozitív, a kisebbeket pedig negatív címkével láttuk el. Ezen kategóriák aránya látható a fenti ábrán.

A korpusz automatizált feldolgozása érdekében még kisebb adatmanipulációs műveleteket is szükséges volt elvégeznünk annak érdekében, hogy a feldolgozásra kialakított pipeline megfelelően tudjon működni. A mondat végén alkalmazott rövidítések gyakran rossz döntésre késztették a mondathatár elválasztásért felelős HunToken eszközt, ezért hozzáadtunk „.”-ot a mondat végéhez. Hasonló problémával szembesültünk, amennyiben a mondat első karaktere kisbetűs volt, ezért azokat nagybetűssé alakítottuk.

[1] http://www.opendata.hu/storage/f/2016-06-06T11%3A27%3A11.366Z/precosenti.zip

3.2. Morfológiai elemzés és egyértelműsítés

A nyelvfeldolgozó pipeline-t a 3 ábra szemlélteti. Tokenizáláshoz a HunTokent használjuk, amelynek xml kimenetét plain textté alakítjuk és az üres sorok eltávolítása után a HunPos [14], illetve a Hunmorph [15] segítségével szófaji és morfológiai elemzést végzünk. A morfológiai egyértelműsítést a szófaji címkéket felhasználó heurisztika alapján végezzük.



3. ábra. Morfológiai elemzés folyamata

Amíg a HunPos figyelembe veszi annak mondatbeli kontextusát, ezért egyértelmű értéket rendel minden egyes szóhoz, addig a HunMorph az egyes szavak összes lehetséges morfológiai felbontását adja kimenetül, ezért egy morfológiai egyértelműsítő implementálására volt szükség. Utóbbi megvizsgálja, hogy hány és milyen kimenettel rendelkezik a HunMorph, majd kiválasztja a HunPos szófajának megfelelő kimenetűt. Előfordulhat, hogy több megoldás is létezik, ilyenkor az elsőt választjuk. Az így előállított kimeneten egy paraméter segítségével állítható, hogy a szótövet vagy a szófajt is tartalmazó alakot használjuk fel betanításra.

3.3. Utószűrés

Az előkészített tokeneken már elvégezhető lenne az elemzés, azonban előtte még kisebb utószűrési feladatok elvégzését láttuk célszerűnek. A funkciószavak (stopwords) szűrésen felül, a számok tanító halmazból történő eltávolítása is hasznosnak bizonyult a szentiment elemzés szempontjából. Továbbá a nagyon ritkán – jelen esetben háromnál kevesebbszer – előforduló kifejezések egy új, eddig nem létező tokennel kerültek helyettesítésre.

3.4. Tulajdonságkinyerés és felügyelt gépi tanulás

A rendelkezésre álló adatok ritkaságát figyelembe véve és a számítási kapacitás csökkentése érdekében úgy döntöttünk, hogy a következő, kizárólag unigram alapú tulajdonságokat fogjuk alkalmazni a modell betanítása során:

Szimmetrikus n szó széles ablak. A korpuszban több olyan mondat is szerepel, melyben több különböző entitást is tartalmaz, ezért azok környezetét kiemelt jelentőséggel kezeltük a betanítás során. Ennek érdekében egy entitás körüli szimmetrikus n széles ablak alkalmazása mellett döntöttünk. A legjobb konfiguráció 5 széles kontextust vesz figyelembe a szó előtt és után is.

Szavak előfordulása TFIDF szerint súlyozva. A szentiment elemzés során nagyon gyakran alkalmazott módszer, mellyel a gyakran előforduló szavak kicsi, míg a ritkábban előforduló kifejezések magasabb súllyal vesszük számításba. Ezáltal a véleménykinyerés szempontjából fontos kifejezéseket magasabb értékkel szerepeltetjük. Erre a célra az Sklearn TFIDFTransformer függvényét használtuk lineáris TF és smooth IDF paraméterekkel. Utóbbit a nullával történő osztás elkerülése végett alkalmaztuk.

Szentiment szótárakban előforduló szavak száma. A modell pontosságának javításán túl a valós életben történő használhatóságot is figyelembe véve, célravezetőnek véltük előre elkészített szentiment szótárak alkalmazását. A PrecoSenti pozitív és negatív polaritáslexikonok külön-külön tulajdonságként kerültek implementálásra, s feleakkora súllyal lettek figyelembe véve. Előbbi 1748, utóbbi 5940 kifejezést tartalmaz.

Az optimális modell elkészítése érdekében több különféle osztályozó algoritmust is kipróbáltunk, mint az SVM több különböző kernellel, multinomiális NaiveBayes és a logisztikus regresszió. Az optimális paraméterek megválasztását is automatizálva végeztük, az Sklearn GridSearchCV funkció segítségével, tízszeres keresztvalidációval. Az elkészített modell a független teszthalmazon került kiértékelésre.

4. Eredmények

A feladat megvalósítása kapcsán arra törekedtünk, hogy a rendszer ne csak az arany sztenderként alkalmazott OpinHuBank korpuszon, hanem lehetőleg valós körülmények között is alkalmazható legyen, így a modell fejlesztése és tesztelése kapcsán több valós példán is tesztelést végeztünk.

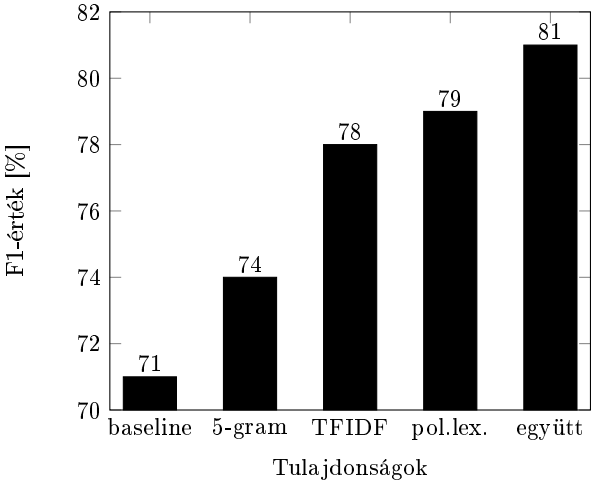
4.1. Diszkusszió

Mivel a szövegbányászati modell fejlesztése iteratív feladatnak számít, az ideálisnak vélt tulajdonságok, gépi tanuló algoritmus és paraméterek kiválasztását csak több teszt futtatása után tudtuk meghatározni. A kiértékeléshez pontosságot (precision), fedést (recall) és F1-mértéket (F1 score) használtunk.

Első körben egy háromosztályos véleménydetekciót végeztünk, azonban a kiértékelés során mért 66% feletti F1-mérték ellenére, a túl nagy számban jelenlévő semleges vélemények miatt a valós életbeli példákon nagyfokú torzítás jelentkezett. Emiatt a semleges teszthalmaz eltávolítása mellett döntöttünk, s ilyen módon is részletes vizsgálat alá vetettük az eredményeket. Ezúttal azt tapasztaltuk, hogy a modell kiértékelése során kapott F1-mérték nagyjából megegyezik a valós életből vett mintapéldákra letesztelt eredményekkel.

A felügyelt gépi tanuló algoritmus közül a kifejezetten szövegelemzési célra fejlesztett multinomiális Naive Bayes alkalmazás bizonyult célravezetőnek, azonban csak minimális, körülbelül 1%-kal jobb eredményt biztosított, mint az SVM algoritmus lineáris kernellel vagy a logisztikus regresszió. Ezzel szemben a korpusz megfelelő előfeldolgozásával, és a tulajdonságkinyerés segítségével jelentős pontosságnövekedést értünk el.

A fenti ábrán szemléltetésre kerültek a kétosztályos esetre elkészített tulajdonságok alkalmazása külön-külön, illetve együttes alkalmazásának hatásai a kiértékelés során mért F1-mértékre. A baseline rendszer, azaz a szimpla unigram



4. ábra. Tulajdonságkinyerés hatása a kétosztályos feladat során

alapú szószámlálást felhasználva elért 71%-os érték jelentősen, 10%-kal növelhető az imént bemutatott jellemzők együttes felhasználásával.

1. táblázat. Legjobb kétosztályos entitásorientált modell kiértékelése

Címke	Pontosság	Fedés	F1-mérték	Teszt bejegyzések száma
negatív	0.78	0.90	0.84	460
pozitív	0.85	0.71	0.77	388
átlag/össz	**0.82**	**0.81**	**0.81**	**848**

A legjobb eredményt fent ismertetett tulajdonságok együttes használata, és a multinomiális Naive Bayes osztályozó következő paraméterei szolgáltatták: alpha: 0.75, class_prior: None, fit_prior: False.

4.2. Eredmények összehasonlítása

Az betanított modell elkészítését követően fontosnak tartottuk annak összehasonlítását meglévő magyar nyelvre implementált megoldásokkal, amelyek kiértékeléssel is rendelkeznek. Az itt bemutatott munkához leginkább a [9] hasonló, amely szintén entitásorientált megközelítést alkalmaz az OpinHuBank korpuszt használja.

A Szegedi Tudományegyetem csapata által elkészített kétosztályos megoldás eredményeihez hasonlítjuk munkánkat. A két koncepció már a korpusz előfeldolgozásánál eltér, mivel ők a nem egyértelmű, azaz a pozitív és negatív értékeléssel is rendelkező entitás-mondat párokat nem vették figyelembe, addig mi az összesített pontszámokat vettük figyelembe és csak a 0 összegűeket dobtuk el.

Az általuk elért legjobb eredmény során kizárólag unigram jellemzőket alkalmaztak, meglepő módon a bigram jellemzők rontottak a modell pontosságán. Tulajdonságként nem csupán az entitások közvetlen környezetét vették figyelembe, hanem azoknak az entitásokhoz vett relatív pozíciója alapján történő súlyozását is. Továbbá alkalmaztak előre elkészített szentiment szótárakat is. Így végül 88,5%-os pontosságot (precision) értek el kétosztályos entitás-orientált szentiment elemzés esetén.

Ugyan a mi legjobb konfigurációnk pontosságban elmarad, azonban szabadon elérhető a forráskód, illetve „dobozos termékként” a Docker image.

A tesztadatokat a Polyglot sentiment analysis magyar moduljával is felcímkéztük. A Polyglot háromféle választ ad: pozitív, negatív és nem meghatározott (cannot determine). A tesztadatok 32%-ára adott nem meghatározott választ, a maradék adaton 69%-os pontosságot ér el, a nem meghatározottakat hibásnak számolva a pontosság csupán 46%.

5. Hibaelemzés

A tesztadatokon végeztünk kézi hibaelemzést, amely során az alábbi hibaosztályokat állapítottuk meg a 154 hibásan osztályozott entitásnál: negálás, kétértelműség, szentiment szótár hibája, adatritkaság (a szótárral nem volt átfedés). A 2. táblázat szemlélteti a hibák gyakoriságát.

2. táblázat. Az egyes hibaosztályok gyakorisága

Hibaosztály	Előfordulás	%
negálás	16	10%
kétértelműség	18	12%
szótár	31	20%
adatritkaság	89	58%

A hibaosztályokat példákkal és magyarázattal szemléltetve:

negálás *Az Országos Igazságszolgáltatási Tanács (OIT) kedden úgy döntött, hogy nem támogatja Baka András főbírói jelölését.*

kétértelműség *Azt azért rendesen röhögöm, hogy a képen minden fordítva van, mint a pártéletben: Fodor a hatalmas, Orbán a törpe, és erre még ráerősít a kép torzítása is.* – az „Orbán a törpe" kétértelmű,

szentiment szótár hibája *A norvég politikusok már feladták a reményt, hogy saját védelmi miniszterük, Kristin Krohn Devold nyerje el a tisztséget.* – a remény pozitív szóként szerepel a szótárban,

adatritkaság *Az ír kormányfő biztosította támogatásáról Orbán Viktort.* – a kormányfő szó nem szerepelt a szótárban, a tanítóadatban többször szerepelt negatív kontextusban.

6. Alkalmazás

A feladat kitűzésekor a valós életben használható modell betanításán felül egy olyan alkalmazás elkészítésére helyeztük a hangsúly, mely bárki számára elérhető, platformfüggetlenül és intuitív módon használható. Előbbi érdekében a telepítő elkészítésén túl létrehoztunk egy előre inicializált Docker containert[2], míg utóbbi érdekében egy REST API hozzáférést nyitottunk.

A Docker image a bemutatott teljes pipeline-t előre telepítve tartalmazza. Az elemezni kívánt szövegrészletet a REST API-n keresztül Windows esetén külön alkalmazásból (példaképp WizTools[3]), Linux vagy Mac OS X operációs rendszernél pedig akár a parancssorból a következőképpen lehet beküldeni:

```
curl -i -H "Content-Type: application/json" -X POST -d
'{"sentence": "Ide írja a az elemezni kívánt szöveget."}'
http://172.0.0.1:5000/sentiment_verbose
```

A predikció során részletes eredményeket közlünk, azaz a teljes szövegrészlet szentimentjén felül az egyes entitásokra kapott pozitív szentiment valószínűségét is megadjuk. Ezek alapján egy harmadik, semleges kategória is kialakításra került, ha a két eredmény között kisebb, mint 15% a különbség. Az entitások és azok kategóriájának (tulajdon, földrajzi és cégnév) meghatározására a Polyglot NER modulját alkalmaztuk, véleményelemzésre pedig az azok körüli szimmetrikus 5 széles kontextusablak alkalmazása után, a szűkített adathalmazon kerül sor. A 5 ábrán egy ilyen részletes elemzésre adunk egy példát.

Az alkalmazás részletes használati útmutatóját, forráskódját, telepítőjét nyilvánosságra hoztuk a GitHubon.[4]

Hivatkozások

1. John Pavlopoulos Haris Papageorgiou Ion Androutsopoulos Suresh Manandhar Maria Pontiki, Dimitrios Galanis. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, SemEval '14, pages 333–352, Dublin, Ireland, 2014. Association for Computational Linguistics.

[2] https://hub.docker.com/r/dhuszti/sentanalysis/
[3] https://github.com/wiztools/rest-client
[4] https://github.com/dhuszti/SentimentAnalysisHUN

```
{
  "results": [
    {
      "input sentence": "A bajnok csapatból Stephen Curryt választották az elmúlt év
legjobb játékosának. Az ellenfél legjobb játékosa LeBron James sérülése miatt sajnos
nem játszhatott a döntőben.",
      "negative probability": 0.10785739058451087,
      "positive probalitiy": 0.89214260941548962,
      "sentiment": "positive"
    },
    {
      "entity": "Stephen Curryt",
      "entity type": "person",
      "negative prob": 0.15265205159468487,
      "positive prob": 0.84734794840531602,
      "sentiment": "positive"
    },
    {
      "entity": "LeBron James",
      "entity type": "person",
      "negative prob": 0.88357112376108005,
      "positive prob": 0.11642887623891852,
      "sentiment": "negative"
    }
  ]
}
```


5. ábra. Példa az alkalmazás részletes kimenetére

2. Haris Papageorgiou Suresh Manandhar Ion Androutsopoulos Maria Pontiki, Dimitrios Galanis. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval '15, pages 486–495, Denver, Colorado, 2015. Association for Computational Linguistics.
3. Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeny Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, pages 19–30, San Diego, California, June 2016. Association for Computational Linguistics.
4. Josef Steinberger, Tomáš Brychcın, and Michal Konkol. Aspect-level sentiment analysis in czech. *ACL 2014*, page 24, 2014.
5. Meishan Zhang, Yue Zhang, and Duy-Tin Vo. Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on EMNLP*, pages 612–621, 2015.
6. Duy-Tin Vo and Yue Zhang. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pages 1347–1353, 2015.
7. Váradi Tamás Miháltz Márton. Trendminder: politikai témájú facebook üzenetek feldolgozása és szociálpszichológiai elemzése. In *XI. Magyar Számítógépes Nyelvi Konferencia*, MSZNY 2015, pages 195–198, Szeged, Magyarország, Januar 2015. Szegedi Tudományegyetem.
8. Fülöp Éva Kővágó Pál Miháltz Márton Váradi Tamás Pólya Tibor, Csertő István. A véleményváltozás azonosítása politikai témájú közösségi médiában megjelenő szövegekben. In *XI. Magyar Számítógépes Nyelvi Konferencia*, MSZNY 2015, pages 198–209, Szeged, Magyarország, Januar 2015. Szegedi Tudományegyetem.
9. Berend Gábor Hangya Viktor, Farkas Richárd. Entitásorientált véleménydetekció webes híranyagokból. In *XI. Magyar Számítógépes Nyelvi Konferencia*, MSZNY 2015, pages 227–234, Szeged, Magyarország, Januar 2015. Szegedi Tudományegyetem.
10. Miháltz Márton. Opinhubank: szabadon hozzáférhető annotált korpusz magyar nyelvű véleményelemzéshez. In *IX. Magyar Számítógépes Nyelvi Konferencia*,

MSZNY 2013, pages 343–345, Szeged, Magyarország, Januar 2013. Szegedi Tudományegyetem.

11. Yanqing Chen and Steven Skiena. Building sentiment lexicons for all major languages. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers)*, pages 383–389, 2014.
12. Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. Polyglot-NER: Massive multilingual named entity recognition. *Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, British Columbia, Canada, April 30 - May 2, 2015*, April 2015.
13. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
14. Peter Halacsy, Andras Kornai, and Csaba Oravecz. HunPos: an open source trigram tagger. In John Carroll and Eva Hajicova, editors, *Proc. ACL 2007 Demo and Poster Sessions*, pages 209–212. ACL, Prague, 2007.
15. Viktor Trón, György Gyepesi, Péter Halácsky, András Kornai, László Németh, and Dániel Varga. Hunmorph: Open source word analysis. In *Proceedings of the ACL Workshop on Software*, pages 77–85. Association for Computational Linguistics, Ann Arbor, Michigan, 2005.

A szentimentérték módosulásának vizsgálata szemantikai–pragmatikai szempontból annotált korpuszon

Szabó Martina Katalin[1,2], Nyíri Zsófi[1], Morvay Gergely[1], Lázár Bernadett[1]

[1] Precognox Informatikai Kft.
{mszabo,zsnyiri,gmorvay,blazar}@precognox.com
[2] Szegedi Tudományegyetem, Bölcsészettudományi Kar,
Szláv Intézet
szabo.martina@lit.u-szeged.hu

Kivonat: A dolgozat az emotív szemantikai tartalmú elemek egy speciális csoportját vizsgálja, kézzel annotált korpusz segítségével. Negatív emotív szemantikai tartalmú elemekként hivatkozunk azokra a kifejezésekre, amelyek képesek arra, hogy elsődleges negatív polaritásuk ellenére pozitív értékelést fejezzenek ki, vagy (pozitív és negatív) szentimentkifejezések fokozóiként szolgáljanak. A vizsgálat tárgyát képező elemek – az elméleti szempontú problematikusságuk mellett – nyelvtechnológiai szempontból is figyelemre méltóak. Automatikus kezelésük ugyanis komoly kihívást jelent mind a szentiment-, mind az emócióelemzés számára. A vizsgálati korpuszt, amelyet a kutatás céljainak megfelelően annotáltunk kézzel, magyar nyelvű twitter-bejegyzések alkotják. A korpusz létrehozásának fő célja egy olyan adatbázis megalkotása volt, amely lehetővé teszi az adott elemcsoport beható, szemantikai–pragmatikai szempontú vizsgálatát. A dolgozatban beszámolunk a korpusz létrehozásának menetéről és eszközéről, az annotálás alapelveiről, valamint a korpuszadatok vizsgálati eredményeiről is. Végül összegezzük azokat a javaslatokat, észrevételeket, amelyek figyelembe vétele véleményünk szerint hozzásegíthet a vizsgált elemek pontosabb automatikus feldolgozásához.

1 Bevezetés

A dolgozat egy az elméleti nyelvészetben is kevéssé tárgyalt, és a nyelvtechnológia számára is problematikus jelenséget, az ún. negatív emotív szemantikai tartalmú elemeket veszi górcső alá.

A vizsgált jelenség a szentimentelemzés egy kardinális részproblémájához, a polaritás, másképpen a szentimentérték módosulásához tartozik. A szentimentérték módosulásának azt a jelenséget nevezzük, amikor egy adott nyelvi elem lexikai szintű szentimentértéke nem azonos, vagy nem teljes mértékben azonos az őt magában foglaló teljes megnyilatkozás értékével [1].

Egy adott szentimentkifejezés lexikai szintű polaritása számos okból kifolyólag eltérhet a bennfoglaló megnyilatkozás polaritásától. Így például, egy pozitív kifejezés polaritása negálható (pl. *nem jó*), vagy bizonytalanná tehető (pl. *kevésbé jó*), vagy az irónia eszközével az ellentétére fordítható (pl. *ezt jól megcsináltad!*).

A szentimentérték módosulásának egy speciális típusát képezik azok az elemek, amelyek lexikai szinten negatív emotív szemantikai tartalommal rendelkeznek ugyan, azonban szenti-

mentkifejezés funkcióját betöltve, vagy pedig más szentimentkifejezések fokozóiként képesek nem negatív értékítélet kifejezésére is [2,3]. E kifejezéseknek két altípusát különböztetjük meg: az ún. *lexikai szintű értékváltás*t (1a) és az ún. *értékvesztés*t (1b) (a vizsgált elemeket kövér szedéssel emeljük ki):

1. a. **brutális** koncerten voltunk a hétvégén
   b. **brutálisan** jó volt a tegnap esti buli

Az (1a) alatti példa esetében azt látjuk, hogy a vizsgált elem aktuális polaritása ellentétes azzal, mint amelyet lexikai szinten hordoz. Ezt a jelenséget nevezzük *lexikai szintű értékváltás*nak. Az (1b) alatti példában ugyanakkor a vizsgált elem egy másik, pozitív polaritással rendelkező kifejezés fokozójának (intenzifikálójának) a funkcióját tölti be. Saját, lexikai szintű negatív értékét tehát elveszíti, és egy másik szentimentkifejezés polaritását erősíti tovább. Ezt a jelenséget nevezzük *értékvesztés*nek.

Amint arra a következő fejezetben rámutatunk (l. lentebb, 2.), a vizsgált jelenségekkel mind elméleti, mind alkalmazott nyelvészeti, különösen nyelvtechnológiai szempontból csekély számú dolgozat foglalkozik. Ugyanakkor azok tárgyalása mindkét tudományterületen fontos volna. Ami az elméleti vonatkozásokat illeti, amint azt tárgyalni fogjuk, a vizsgált elemek szemantikai viselkedéséről nincs egységes vélekedés. Ami az alkalmazott nyelvészeti, köztük a nyelvtechnológiai alkalmazásokat illeti, a polaritás módosulásának ezeket a típusait a jelenleg legelterjedtebbnek tekinthető szótáralapú szentimentelemzéssel nem lehet kezelni. A lexikai szintű értékváltásra, illetve értékvesztésre képes nyelvi elemek ugyanis rendszerint szerepelnek a szótári polaritásuknak megfelelő szentimentlexikonban. Ennek következtében automatikus kezelésük a szótáralapú elemzés során tévesen történik. Ugyancsak problematikusak a automatikus emócióelemzés szempontjából is, hiszen a szótáras elemzés ezeket a kifejezéseket azok lexikai szintű negatív tartalma alapján azonosítja (részletesebben l. [4]).

A jelen dolgozatban bemutatjuk azt a kézzel annotált korpuszt, amelyet specifikusan e probléma vizsgálatára hoztunk létre. Ismertetjük továbbá mindazokat a vizsgálati eredményeket, amelyeket a nyelvhasználati sajátságokat illetően megállapítottunk, a korpuszadatok feldolgozása és elemzése alapján. Végezetül azt is tárgyaljuk, milyen lehetőségeket látunk a vizsgálat tanulságainak nyelvtechnológiai implementációjára.

2 Szakirodalmi áttekintés

Az általunk lexikai szintű értékváltásnak, valamint értékvesztésnek nevezett jelenségekkel foglalkozó dolgozatok többsége a pszichológia, valamint az elmélet oldaláról teszi vizsgálat tárgyává a problémát [5,6,7,8,9,10]. Ami a szentimentelemzéshez kapcsolódó nyelvtechnológiai kutatásokat illeti, megállapítható, hogy amíg az ún. kontextuális polaritásváltással (pl. a negáló elemek problémája) egyre gyakrabban foglalkoznak az irodalmi tételek, addig az általunk vizsgált jelenségekre csupán csekély számú dolgozat fordít figyelmet [2,3,11].

Andor [9] alapján a jelen dolgozatban tárgyalt jelenség nem ritka a jelentésváltozások folyamataiban. Magyarázata szerint annak „leggyakoribb eseteiben a negatív jelentéstartalmú és használatú lexikális egységek pozitív irányú jelentésváltozását vagy jelentésbővülését, jelentésük kiterjesztését figyelhetjük meg” [9]. A szerző véleménye szerint a jelenség különösen „az értékítéletet, fokozást kifejező ún. intenzifikáló szavak körében jellemző”. Ugyanakkor arra is felhívja a figyelmet, hogy ezek az elemek gyakran kollokálódnak negatív polaritású melléknevekkel is amellett, hogy egyre nagyobb arányban fordulnak elő pozitív tartalmú kifejezésekben, konstrukciókban.

Andor [9] és Kugler [10] is megemlíti, hogy a jelenség az intenzifikáló elemek kapcsán az ellentétes polaritás irányába is lehetséges, azaz pozitívból negatívba (pl. tökéletesen buta, jól elhibázta). Ugyanakkor véleményük szerint ez utóbbi jóval ritkább előfordulású.

Tolcsvai Nagy [5] amellett érvel, hogy amíg az általa „hagyományosnak" nevezett jelzői csoport tagjai, így például a *kiváló*, a *nagyszerű* vagy a *csodálatos* elemek magukban hordozzák „a szóval jelölt érték valódiságát", addig az *állati* vagy a *baromi* típusú jelzők „a jelzett értékek relativitására utalnak". Ennek következtében azok jelentésében „ironizáló magatartás" mutatkozik meg. Tolcsvai Nagy [5] érvelésével ellentétben Székely [12] ugyanakkor úgy véli, hogy a vizsgált elemek szemantikailag gyakorta motiválatlanok.

A jelenség Andor [13] és Jing-Schmidt [7] vizsgálati eredményei alapján számos nyelvben megtalálható, ezért valószínűleg nyelvfüggetlen sajátság.

Kugler [10] és Jing-Schmidt [7] a jelenséget elsősorban a használat lehetséges pszichológiai oka szempontjából vizsgálja. Ennek kapcsán Kugler [10] felhívja a figyelmet a kongruencia, másképpen értékbeli egyez(tet)és tendenciájára, miszerint „a legtermészetesebb és ezért a legkönnyebben feldolgozható szerkezetekben azonos polaritású kifejezések kapcsolódnak össze. Amennyiben a kifejezés tagjai között nincs kongruencia, a szerkezet nagyobb mentális erőfeszítéssel (így szükségképpen hosszabb idő alatt) dolgozható fel. Véleményünk [11] illeszkedik Kugler [10] fentebbi megállapításához: az inkongruencia a vizsgált jelenség egyik pszichológiai motiválójának tekinthető, hiszen az így egymás mellé kerülő, ellentétes polaritású tartalmak interpretálása – az említett tendencia miatt – nagyobb figyelmet igényel a hallgatótól.

3. A vizsgálati korpusz létrehozásának és feldolgozásának a menete

A nyers korpusz, amelyből a kutatáshoz szükséges anyagot gyűjtöttük, összesen 37818 magyar nyelvű twitter-bejegyzésekből áll. A jelen kutatáshoz a korpusz azon nyelvi adataira volt szükségünk, amelyek tartalmaztak legalább egy, lexikai szintű értékváltásra vagy értékvesztésre képes elemet. A munka első fázisában ki kellett tehát nyernünk a korpuszból az ennek a szempontnak megfelelő tweeteket. A feladathoz kézzel összeállítottunk egy listát, amely lexikai szintű értékváltásra vagy értékvesztésre képes elemeket tartalmaz. A munkában egy fokozó értelmű kifejezéseket tartalmazó szótárra [14], két korpuszra [15,16], valamint internetes adatokra támaszkodtunk. Az így létrehozott szólista 109 szóalakot tartalmaz. Ezt követően automatikus módszerrel kigyűjtöttünk a korpuszból minden olyan tweetet, amely tartalmazott legalább egyet a listánkban szereplő elemek közül. Az így létrehozott korpusz összesen 610 tweetből áll.

A nyers korpusz kézi feldolgozásához a Brat nevű, online elérhető annotáló programot használtuk [17]. Az eszköz bármilyen annotálási feladatra alkalmas, a felhasználó maga konfigurálhatja a használni kívánt annotációs tageket, azok csoportjait, kapcsolatatait, és további, bevinni kívánt információkat. Teljesen személyre szabható és egyszerűen kezelhető. A config fájlt és az annotálandó szövegeket txt formátumban töltöttük fel a programba, és az annotációs fájlokat is ebben a formátumban kaptuk vissza. Az output fájlban az annotált tagek lokációi és a létrehozott kapcsolatok szerepelnek egyszerű listaként.

Az adatbázis feldolgozását a következő annotációs szempontok alapján végeztük el manuálisan: Először bejelöltük azokat az elemet, amelyek esetében a lexikai szintű szentimentérték eltért a kontextusbeli értéktől. Még ebben a lépésben döntést hoztunk arról is, hogy ez az eltérés az aktuális kontextusban lexikai szintű értékváltásnak, vagy pedig értékvesztésnek köszönhető-e, és ennek megfelelően jelöltük a vizsgált elemet, valamint a targetét vagy azt, amit módosít, tehát a frázis alaptagját. A lexikai szintű értékváltás (2a) esetében az előbbit, az értékvesztés (2b) esetében az utóbbit kerestük meg és annotáltuk. (A vizsgált elemeket ebben az esetben is kövér szedéssel, míg a további annotált kifejezéseket aláhúzással jelöljük.)

2. a. ismét **brutális** koncertet adott az énekes
   b. Valljátok be, **rohadt** jó az időérzékem!

Amint a példák mutatják, a (2a) esetében a vizsgált elem, lexikai szintű negatív polaritását elveszítve pozitív szentimentkifejezés funkcióját tölti be, és a *koncert* targetre vonatkozóan fejezi ki ezt a pozitív értékelést. Ettől eltérően, a (2b) alatti példában a vizsgált elem nem szentimentkifejezésként funkcionál, hanem a szintaktikai szerkezetben az alaptag szerepét betöltő szentimentkifejezés fokozójaként, annak szemantikai tartalmát erősítve.

A bemutatott annotálási megoldásnak az volt a célja, hogy a segítségével a jövőben vizsgálni lehessen egyrészt a fokozó szerepű, értékvesztésre képes elemeknek és az általuk módosított elemeknek, valamint a lexikai szintű értékváltásra képes elemeknek és azok targeteinek a kapcsolatait.

Az annotációban értelemszerűen csupán azokat a tweeteket annotáltuk, ahol a vizsgált elemnél lexikai szintű értékváltást vagy értékvesztést véltünk felfedezni. Azokban az esetekben tehát, ahol a vizsgált elem aktuális polaritása megegyezett annak elsődleges, azaz negatív polaritásával, nem annotáltuk, pl.

3. Egy **rohadt** kukásautó miatt áll már vagy tíz perce a troli.

A fentebb bemutatott két alaptípus annotálásán túl az értékvesztésre vonatkozóan további tageket is bevittünk, az annotált elemek további szemantikai–pragmatikai viselkedése alapján. E megoldás alkalmazása mellett korábbi vizsgálati eredményeink alapján döntöttünk: megfigyeltük, hogy amíg a negatív emotív tartalmú fokozó elemek pozitív és negatív polaritású kifejezések módosítóiként rendre deszemantizálódnak (4a-b), addig semleges melléknévi alaptagok mellett változatos szemantikai viselkedést mutatnak (4c-d) [11]. (A példákban a vizsgált elemet kövérrel, az alaptagot aláhúzással jelöltem.)

4. a. **iszonyat** jó volt ez a 4 nap, köszönöm az élményt
   b. **borzasztó** unalmas Affleck a szerepben
   c. 166 centi 46 kiló az milyen egy 14 éves lánynak? Nem vagyok **rohadt** magas? (vö. *túl*)
   d. A processzor teljesítménye elégnek tűnik, mert minden **marha** gyors

A (4c-d) alatti példák arra mutatnak, hogy semleges alaptag mellett a fokozó elem, negatív szemantikai tartalmát illetően nem feltétlenül üresedik ki, és válik puszta fokozóvá. A(4c-d) alatti tweetekben ugyanis éppen ez az elem adja hozzá a negatív értékelést a szöveghez, ellentétben a (4a-b) alatti példákkal, ahol negatív tartalommal nem számolhatunk.

Annak céljából, hogy ezeket az eseteket a korpusz alapján vizsgálni tudjuk, az intenzifikáló elemek esetében azt is annotálnunk kellett, hogy az általuk módosított elemek pozitív vagy negatív polaritásúnak, vagy pedig semlegesnek tekinthetőek-e, továbbá, hogy az intenzifikáló elemek milyen aktuális szemantikai tartalommal rendelkeznek.

Mindemellett, az összes annotált taghez létrehoztunk egy “egyéb”-kategóriát is azért, hogy a munka során esetlegesen felmerülő, sajátos jelenségeket is jelölni tudjuk egy későbbi vizsgálat céljából.

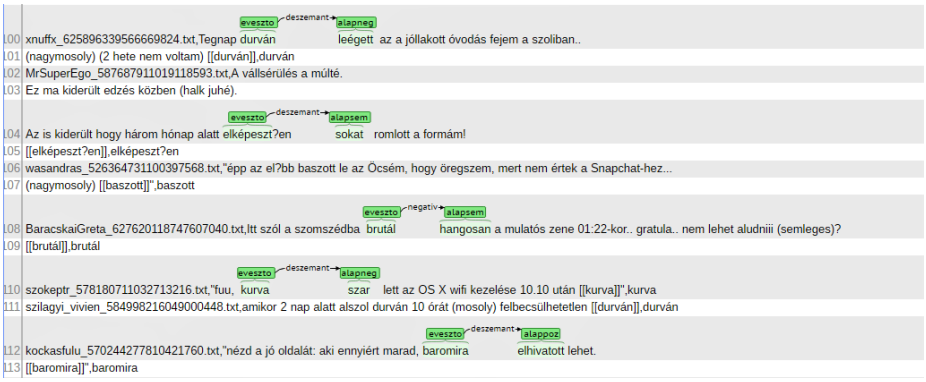
A fentebb ismertetett annotálási rendszert az alábbi táblázat foglalja röviden össze:

	Az annotált tagek és szemantikai–pragmatikai sajátságok (ez utóbbit a program technikai sajátsága okán viszonyként jelöltük):
Lexikai szintű értékváltás esetén:	lexikai szintű értékváltó elem, ami a szemantikai tartalma alapján lehetett: értékváltó vagy egyéb
	target
Értékvesztés esetén:	értékvesztő elem, ami a szemantikai tartalma alapján lehetett: deszemantizált, negatív, pozitív vagy egyéb
	alaptag, ami a szemantikai tartalma alapján lehetett: negatív, pozitív, semleges vagy egyéb

1. táblázat. Az annotálási rendszer rövid összefoglaló táblázata.

A bemutatott annotálási rendszert az annotátorok egy részletes annotálási útmutatón keresztül ismerték meg, amelyben az egyes jelenségeket példák segítségével prezentáltuk.

Az alábbi ábra egy részletet közöl a korpusz annotációjából a Brat programban:



1. ábra. Részlet a korpusz annotációjából a Brat nevű programban.

4 Eredmények

4.1 Az egyetértésmérés eredményei

Ahhoz, hogy az annotátorok közötti konszenzust mérni tudjuk, a korpusz feldolgozása előtt elvégeztünk egy egyetértésmérést a korpusz egy kisebb részletén. A méréshez összesen 100 tweetet annotáltunk, tekintettel arra, hogy már ez a mennyiség is a teljes korpusz hatod részét tette ki. Az pilot-adatokon az annotálást követően kappa-statisztikát alkalmaztunk.

A pilot-annotálás megmutatta, hogy az annotátorok a Kappa-érték szerint összesítve 0.489, azaz a Kappa-sávok alapján közepes szintű átlagos egyetértéssel dolgoztak. Az annotáció részletes vizsgálata alapján a közepes szintű eredmény alapvető oka az volt, hogy az annotálás technikai szempontból nem volt egységes: az annotátorok a munka során nem azonos kijelölési megoldásokat alkalmaztak, ami miatt számos lokáció-beli eltérés keletkezett a korpuszban. Ugyanakkor, az eredmény – kisebb részben ugyan, de – összefüggést mutatott az annotálási feladat tartalmi vonatkozásaival is, tehát azzal, hogy nem triviális, hanem szemantikai–

pragmatikai szempontból dolgoztuk fel az adatokat, ami néhol bizonytalanságot eredményezett az annotátorok körében. E fentebbi tapasztalatok alapján a korpusz feldolgozását a technikai megoldások egységesítése, valamint az annotálási alapvelvek pontosítását követően végeztük csak el.

4.2 Az annotált korpusz vizsgálati eredményei

A 610 tweetből összesen 280-at annotáltunk. Ezek voltak azok az esetek ugyanis, ahol a vizsgált elem lexikai szintű értékváltást vagy értékvesztést mutatott. A többi, nem annotált esetben a vizsgált elem megőrizte a lexikai szintű negatív polaritását az aktuális kontextusban.

Az annotációban összesen 41 esetben jelöltünk értékváltást, és 238 esetben értékvesztést, tehát intenzifikálói funkciót.

Megvizsgálva az értékváltás eseteit, a következő megállapításokat tehetjük: A két leggyakoribb elem ebben a szerepben a *durva(-n)* (13 előfordulás) és a *kemény* (8 előfordulás) volt. Számos további elemet is annotáltunk még értékváltóként, azonban ezek lexémánként átlagosan mindössze egy vagy két alkalommal szerepeltek. Szerettük volna megtudni, hogy vajon menynyire jellemző a két leggyakoribb elemre az értékváltás, azaz feltehető-e, hogy ezek az elemek szentimentkifejezés funkciójában alapvetően pozitív aktuális polaritással rendelkeznek. Megvizsgáltuk tehát ezeknek az elemeknek az összes, nem annotált előfordulását is a korpuszban. Azt tapasztaltuk azonban, hogy mindkét elem gyakran fordul elő negatív értékelés kifejezőjeként is. Ez a sajátság nyilvánvalóan megnehezíti az értékváltásra képes elemek aktuális polaritásának a helyes automatikus kezelését (a problémáról részletesebben l. lentebb, 5.).

A 238 értékvesztési esetet illetően a következő észrevételeket tehetjük: Ahogyan azt a feldolgozás menete kapcsán is ismertettük (l. fentebb, 3.), azokban az esetekben, ahol a vizsgált elem fokozó funkciót töltött be, a intenzifikálót és az általa módosított elemet további szemantikai–pragmatikai annotációval láttuk el, és a korpusz felhasználása során az így annotált sajátságokat is lekérdeztük. Az adatokat az alábbi táblázat mutatja be:

		a vizsgált elem aktuális szemantikai tartalma				
		deszem	**negatív**	**pozitív**	**egyéb**	ÖSSZESEN:
az alaptag szemantikai tartalma	**alappoz**	94	-	-	1	95
	alapneg	70	-	-	8	78
	alapseml	30	24	3	-	57
	egyéb	5	-	-	3	8
	ÖSSZESEN:	199	24	3	12	238

2. táblázat. Az annotált értékvesztési esetek részletes statisztikai adatai

Az eredmények alapján a következő megállapításokat tehetjük: A korpusz annotátorai az összesen 238 esetből 199 alkalommal vélték úgy, hogy a vizsgált elem teljesen elvesztette elsődleges

negatív szemantikai tartalmát, és az általa módosított elem mellett puszta fokozó szerepet töltött be. Ez az összes eset 83,61%-át tette ki.

A vizsgált elemek a polaritásvesztést illetően a pozitív alaptagok mellett mutatkoztak a legegységesebbnek. Megállapíthatjuk ugyanis, hogy pozitív alaptagok módosítóiként rendre elveszítik lexikai szintű tartalmukat, l. pl. (4b) fentebb. Ugyancsak kis mértékű eltérést látunk az annotációban akkor is, ha a módosított elem negatív polaritású. Összességében úgy tűnik tehát, hogy deszemantizálódnak azok a fokozó értelmű elemek, amelyek valamilyen (pozitív vagy negatív) polaritással rendelkező elemet módosítanak.

A fentebb bemutatottakkal ellentétben, semleges alaptagok módosítóiként a vizsgált elemek jelentős szemantikai változatosságot mutatnak. Az 57 eset valamivel több, mint a felében (52,63%) jelöltek az annotátorok deszemantizáltságot, azaz ítélték úgy, hogy a vizsgált elem teljesen elveszíti lexikai szintű negatív értékét. (A módosított elemeket itt is aláhúzással jelöltem.)

5. a. Jáááj, de **rettenetesen** kíváncsi lettem!
   b. Ja és Colin Farrell egy **kibaszott** nagy színészóriás

24 alkalommal (42,1%) azonban a vizsgált elem nem veszítette el negatív szemantikai tartalmát. Ezekben az esetekben tehát, annak ellenére, hogy fokozó funkciót tölt be, az aktuális kontextusban ugyanúgy negatív értékelést fejez ki, mint lexikai szinten, pl.

6. **Szörnyen** meleg van még így az éjszaka közepén is.

Végezetül, 3 olyan esetet is jelöltek a korpusz annotátorai, ahol a vizsgált elemet pozitív polaritás hordozójának értékelték, annak lexikai szintű negatív polaritása ellenére, pl.

7. mondjuk **rohadt** sok programom lesz ebben a hónapban is, de fel nem bírom fogni, hogy utána újra suli

A vizsgálat fentebbi tanulságai azért is figyelemre méltóak, mert számos, a negatív emotív szemantikai tartalmú fokozó elemekkel foglalkozó dolgozat amellett érvel, hogy azok szemantikailag rendre kiüresednek, elveszítik lexikai szintű negatív polaritásukat. Balogh [18] például úgy gondolja, hogy amennyiben „az ilyen, másodlagos fokozóelemeket egy-egy megfelelő kulcsszóhoz kapcsoljuk, elveszítik elsődleges, azaz lexikális jelentésüket és átveszik a fokozó értelmű „nagyon" adverbium jelentését." Hozzá hasonlóan érvel Jing-Schmidt [7], aki szerint a félelem érzelemmel kapcsolatos negatív emóciókifejezések esetében, fokozó szerepben a félelem szemantikai tartalom metonimikusan a magas emotív intenzitásra redukálódik. A (6) alatti tweetben ugyanakkor éppen ez az elem adja hozzá a negatív értékelést a szöveghez, ellentétben az (5) alatti példákkal, ahol nincs ilyen negatív tartalom.

Kíváncsiak voltunk, vajon milyen konkrét fokozó elemek fordulnak elő a korpuszban semleges alaptag mellett, és mutatkozik-e valamilyen eltérés abban, hogy mely elemek üresednek ki szemantikailag, és melyek nem. Megvizsgáltuk tehát mind a deszemantizált, mind a negatív polaritású csoport gyakorisági megoszlásait. A semleges alaptagok melletti deszemantizált fokozó elemek közül a leggyakoribbak a *kurva* (48), a *rohadt* (25), az *iszonyat* (19), a *baromi* (18) és a *(ki)baszott* (17) tövek, illetve ezek különböző alakváltozatai voltak. Megvizsgálva a negatív tartalmú fokozók megoszlását ugyancsak semleges alaptagok mellett azt az érdekes

sajátságot tapasztaltuk, hogy amíg közülük a négy leggyakoribb egybeesik a deszemantizáltak leggyakoribbjaival, addig a 3. leggyakoribb deszemantizált elem, az *iszonyat* negatívként egyetlen egyszer sem fordult elő a korpuszban. Ez az elem tehát minden esetben teljes értékvesztést mutat.

5 A vizsgálati eredmények felhasználhatósága az automatikus szentimentelemzésben

A vizsgált nyelvi jelenség viszonylagosan ritka előfordulása miatt az annotált korpusz mérete kicsinek mondható. Hangsúlyozzuk továbbá, hogy a pilot-korpuszon mért annotátorok közötti egyetértés csupán közepes értéket mutatott, bár – amint azt az egyetértésmérés kapcsán részleteztük (l. fentebb, 4.1) – az eredmény alapvető oka az volt, hogy az annotálás technikai szempontból nem volt egységes, és ezt a problémát orvosoltuk. Az elmondottakkal összefüggésben mégis a következő, a nyelvtechnológiai implementációra vonatkozó megállapításokat a kutatás jelenlegi szakaszában korlátozott érvényűnek tekintjük.

Az automatikus kezelés szempontjából a legkevésbé problematikusak azok a kifejezések, amelyek pozitív vagy negatív polaritású módosított elemek mellett fokozó szerepben állnak. Azt láttuk ugyanis, hogy ennek az összesen 173 esetnek a 94,79%-ában a vizsgált elem deszemantizált volt. Ez alapján azt mondhatjuk, hogy az emotív intenzifikáló elemek értékvesztése a polaritással rendelkező tagok módosítóként egyszerű reguláris szabályokkal leírható.

Problematikusabbak azonban mind a semleges tagok melletti fokozó elemek, mind pedig azok, amelyeket értékváltónak nevezünk, és pozitív szentimentkifejezés funkciójában állnak. Ezekben az esetekben ugyanis nem tudunk gépi módszerrel olyan egyértelmű kontextuális sajátságokra hagyatkozni, mint amilyenekre a fentebb tárgyalt esetben. Amint arról beszámoltunk (l. fentebb, 4.2.), megpróbáltunk gyakorisági eltéréseket felfedezni a semleges alaptagok mellett megjelenő, eltérő szemantikai sajátságokkal bíró csoportokban, azonban egyetlen kivételtől eltekintve nem találtunk érdemi különbséget: a két csoport leggyakoribb elemei megegyeznek; egyetlen kivételtől eltekintve, az *iszonyat* ugyanis csak deszemantizált változatban fordul elő ebben a pozícióban. Ez utóbbi elem szótári jelentését tehát – ezek szerint – fokozó szerepben nem kell figyelembe venni.

Ami a lexikai szintű értékváltás gyakorisági adatait illeti, a *durva* és a *kemény* a legfrekventáltabb elemek a korpuszban. Ugyanakkor megvizsgálva az összes, nem annotált előfordulásukat is, tehát azokat, ahol e kifejezések nem pozitív aktuális értéket hordoztak, azt tapasztaltuk, hogy mindkét elem ugyanolyan gyakran fordul elő negatív értékelés kifejezőjeként is. Aktuális szemantikai tartalmuk tehát lexikai szinten (így például egy szentimentszótárban) nem rögzíthető. Megfigyeltük ugyanakkor, hogy ezeknél az elemeknél, értékváltó pozícióban a tweetelők nagyon gyakran egészítik ki a szöveges megnyilatkozásaikat olyan emotikonnal, amely az aktuális értékelő jelentésre utal. Úgy gondoljuk, talán éppen azért élnek ilyen gyakran emotikonokkal ezekben az esetekben, mert a viszonylag rövid karakterhosszúságú tweetekben, a polaritásváltásra képes elemek használatakor, a közölni kívánt értékelő tartalmat egyértelművé kívánják tenni. Mindezek alapján úgy véljük, hogy a jelen kutatásban megtalált leggyakoribb értékváltó elemeket az aktuális tweet emotikonjával együtt kezelve megnő a helytálló elemzés esélye. Ezzel kapcsolatban érdemes felhívni a figyelmet [19]-re, akik ugyancsak twitter-szövegeket vizsgálva megállapítják, hogy az irónia automatikus felismerésében az emotikonok kulcsszerepet tölthetnek be.

A kutatás következő lépéseként azt tervezzük, hogy a fentebb tárgyalt sajátságokat szabályokba foglaljuk és alkalmazzuk a szótáralapú automatikus szentimentelemzéssel kombinálva, majd felmérjük, hogy azok javítanak-e, és ha igen, milyen mértékben az elemzés eredményességén. Ehhez a feladathoz rendelkezésünkre áll egy kézzel annotált szentimentkorpusz [1]], valamint egy pozitív és negatív polaritású kifejezéseket tartalmazó szentimentszótár [20]. A

korpuszt először egyszerű szóillesztéses megoldással, a szótár alapján elemezzük, és az eredményt összevetjük a kézi annotációval. Ezt követően az elemzést a szótáras módszer és a jelen munka során feltárt sajátságok kombinációjával is elvégezzük, majd ennek a megoldásnak az eredményességét is összevetjük a korpusz kézi annotációjával. Végezetül megnézzük, javult-e, és ha igen, mennyiben az automatikus elemzés eredményessége az alkalmazott szabályoknak köszönhetően.

6 Összegzés

A dolgozatban a negatív emotív szemantikai tartalmú elemek egy specifikus csoportját, az ún. lexikai szintű értékváltásra, illetve értékvesztésre képes elemeket vizsgáltuk kézzel annotált korpusz segítségével.

A vizsgálati korpuszt magyar nyelvű twitter-bejegyzésekből hoztuk létre úgy, hogy egy az erre a célra összeállított elemlista alapján kigyűjtöttük a vizsgálni kívánt elemeket tartalmazó nyelvi adatokat. Ezt követően a korpuszt egy az erre a célra felkészített eszközzel manuálisan annotáltuk. A munka során bejelöltünk minden olyan szemantikai–pragmatikai sajátságot, amellyel a korpusz későbbi, kutatásbeli felhasználását támogatni tudtuk. Az annotáció elkészülte után a korpuszt felhasználtuk a nyelvi jelenség vizsgálatára, és a feltárt sajátságokat részleteiben, példákkal együtt közöltük. Végezetül, a vizsgálat tanulságaira építve tárgyaltuk azt is, hogyan látunk lehetőséget a tapasztalatok nyelvtechnológiai implementációjára.

Ahogyan azt a dolgozatban több ízben kiemeltük, a vizsgált jelenség megfelelő kezelése bizonyos nyelvtechnológiai alkalmazások szempontjából kiemelkedően fontos volna. Így például, ezek az elemek mind a szentiment-, mind az emócióelemzésben téves következtetéseknek engedhetnek teret. Bár a kutatás alapjául szolgáló vizsgálati anyag kis méretű volt, úgy véljük, a segítségével tett megállapítások és javaslatok hozzájárulhatnak egy pontosabb, hatékonyabban működő tartalomelemző rendszer létrehozásához.[1] Ezzel összefüggésben, a munka további lépéseként tervezzük a megállapításaink és javaslataink nagyobb méretű adatbázison való vizsgálatát, igazolását.

Köszönetnyilvánítás

A jelen kutatás Az Emberi Erőforrások Minisztériuma Új Nemzeti Kiválóság Programjának támogatásával valósult meg.

[1] Természetesen egyet kell értenünk a dolgozat névtelen bírálójával abban, hogy a vizsgált jelenség viszonylagosan ritka előfordulása okán annak hatékony kezelése önmagában nem feltétlenül hoz jelentős javulást egy szentiment- vagy emócióelemző rendszer eredményességét illetően. Ugyanakkor amellett érvelünk, hogy a jelenség frekvenciája doménfüggő, és bizonyos típusú, illetve témájú szövegekben kifejezetten gyakorinak tekinthető. Így például a társalgási stílusréteghez tartozó, technológiai témájú szövegek gyakran élnek vele (pl. különböző elektronikai eszközökről, vagy azokkal kapcsolatban írt blog-, facebook- és twitterbejegyzések stb., pl. ***Brutális*** *választék,* ***durván*** *jó árak; várok egy nagyon-nagyon* ***brutális*** *iPhone 7-et; nagyon* ***durva*** *sportkocsi, kicsi, könnyű és* ***iszonyúan*** *erős* stb.), a szentiment- és emócióelemzés egyik leggyakrabban elemzett szöveganyagát pedig – gazdasági okokból – éppen ezek a nyelvi produktumok alkotják.

Irodalom

1. Szabó M. K., Vincze V.: Egy magyar nyelvű szentimentkorpusz létrehozásának tapasztalatai. In: Tanács A., Varga V., Vincze V., eds.: XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015). Szeged, Szegedi Tudományegyetem (2015) 219−226
2. Szabó, M. K.: A polaritásváltás- és változás kezelési lehetőségei a szentimentelemzésben. Tavaszi Szél konferencia konferenciakötete. Budapest, Líceum Kiadó, Eger és Doktoranduszok Országos Szövetsége (2015a) 629−643
3. Szabó M. K.: A polaritásváltás problémája a szentimentelemzés szempontjából. In: Váradi T., ed.: IX. Alkalmazott Nyelvészeti Doktoranduszkonferencia konferenciakötete. Budapest, MTA Nyelvtudományi Intézet, (2015b) 51−61
4. Drávucz, F., Szabó, M. K., Vincze V.: Szentiment- és emóciószótárak eredményességének mérése emóció- és szentimentkorpuszokon. A jelen kötetben
5. Tolcsvai Nagy G.: A mai magyar nyelv normarendszerének egy jelentős változásáról az „ifjúsági nyelv" kapcsán. Magyar Nyelvőr 112(4) (1988) 398−406
6. Wierzbicka, A.: Australian cultural scripts − *bloody* revisited. Journal of Pragmatics, Volume 34(9) (2002) 1167–1209
7. Jing-Schmidt, Z.: Negativity bias in language: A cognitive-affective model of emotive intensifiers. Cognitive Linguistics 18(3) (2007) 417−443
8. Laczkó M.: Napjaink tizenéveseinek beszéde szóhasználati jellemzők alapján. Magyar Nyelvőr 131(2) (2007) 173−184
9. Andor J.: De durva ez a téma! – Megfigyelések a melléknévi polaritásváltásról. In Hungarológiai Évkönyv 12 (2011) 33−42
10. Kugler N.: A nyelvi polaritás kifejezésének egy mintázata, avagy milyen a félelmetesen jó? Magyar Nyelvőr 138(2) (2014) 129−139.
11. Szabó, M. K.: The usage of elements with emotive semantic content from a gender point of view. Kézirat
12. Székely G.: Egy sajátos nyelvi jelenség, a fokozás. In: Segédkönyvek a nyelvészet tanulmányozásához 66. Budapest, Tinta (2007)
13. Andor J.: Functional Studies in the Polarity and Gradation of Amplifier Adjectives and Adverbs in English. In: Andor, J., Horváth, J., Nikolov, M., eds. Studies in English Theoretical and Applied Linguistics. Pécs, Lingua Franca Csoport (2003) 43–59.
14. Tukacs, T.: Túlzásba vitt szavak. A fokozó értelmű szókapcsolatok magyar angol szótára. Budapest, Tinta (2015)
15. Váradi, T.: 2002. The Hungarian National Corpus. In: Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002) European Language Resources Association, Paris (2002) 385–389
16. Oravecz, Cs., Váradi, T., Sass, B.: The Hungarian Gigaword Corpus. In Proceedings of LREC 2014 (2014)
17. Brat online annotáló eszköz (http://brat.nlplab.org/)
18. Balogh, P.: Gender-markerek a nyelvben (2009) http://webfu.univie.ac.at/wp/565
19. Carvalho, P., Sarmento, L., Silva, M. J., Oliveira, E.: Clues for Detecting Irony in User Gene-rated Contents: Oh...!! It's "so easy" ;-). University of Lisbon, Faculty of. Sciences, LASIGE. (2015)
20. Szabó M.K. 2015.: Egy magyar nyelvű szentimentlexikon létrehozásának tapasztalatai és dilemmái. In Gecső T., Sárdi Cs. (eds.) Nyelv, kultúra, társadalom. Segédkönyvek a nyelvészet tanulmányozásához 177. Budapest, Tinta. pp. 278–285.

V. Többnyelvűség

Négy hatás alatt álló nyelv – Korpuszépítés kis uráli nyelvekre

Simon Eszter

MTA Nyelvtudományi Intézet
1068 Budapest, Benczúr u. 33., e-mail: simon.eszter@nytud.mta.hu

Kivonat Cikkünkben bemutatunk egy pilot projektet, amely azt tűzte ki célul, hogy annotált nyelvi adatbázist épít négy oroszországi kisebbségi uráli nyelvre, melyek az udmurt, a tundrai nyenyec, valamint a szinjai és a szurguti hanti. A célkitűzést többek közt az indokolja, hogy az uralisztika területén inkább eklektikus adathalmazokkal találkozik a kutató, mintsem szisztematikusan annotált adatbázisokkal. Meggyőződésünk, hogy a számítógépes nyelvészet eszköztára jól használható az ilyen speciális nyelvekre történő korpuszépítés során is, és nagyban segíti az uralisták és az elméleti nyelvészek munkáját.

Kulcsszavak: korpuszépítés, számítógépes nyelvészet, uráli nyelvek, veszélyeztetett nyelvek

1. Bevezetés

Az uralisztikai kutatások jellemzően az alábbi séma szerint zajlanak. A kutató terepmunkára megy valahova Oroszországba, hazatér egy adag audió- és/vagy videófájllal, amit később feldolgoz a saját elképzeléseinek és céljainak megfelelően. Az adathalmazon kikutatott eredményeket publikálja, de az adathalmazt nem teszi publikusan hozzáférhetővé. Ha valaki valahogy mégis hozzá tud jutni az adatokhoz, akkor azzal szembesül, hogy a kutató a beszélt nyelvi anyagot valami saját lejegyzési rendszer alapján jegyezte le, amit rajta kívül senki nem használ, és nem is ismer. Dokumentáció, ami alapján meg lehetne fejteni a kódot, általában nincs, ha mégis van, akkor nincs publikálva, ha mégis, akkor nem angolul. A lejegyzés szerencsés esetben egy általánosan használt, szabadon elérhető eszközzel történik, de sokszor inkább különféle szövegszerkesztőkben, különféle házilag készített fontkészletekkel összeeszkábált, a strukturáltságnak látszatát kelteni sem igyekvő dokumentumok születnek. Ehhez jön hozzá, hogy a felvételek jogi háttere sokszor nem tisztázott, így a felhasználási lehetőségük is eléggé korlátozott.

Az elmúlt néhány évben/évtizedben a fentiekben vázolthoz képest pozitív változások zajlanak általában véve a nyelvi dokumentáció terén, szűkebben pedig az uralisztikában is. Egyre többen törekszenek arra, hogy szabadon hozzáférhetővé tegyék az adataikat, hogy sztenderd eszközöket használjanak, és hogy

valamilyen formában alkalmazzák a számítógépes nyelvészet eszközeit és/vagy módszereit ahhoz, hogy ne egy eklektikus adathalmazt, hanem egy strukturált adatbázist kapjanak eredményül.

Cikkünkben egy olyan projektet mutatunk be, amely szintén ezt a célt tűzte ki, vagyis egy nyelvi annotációt tartalmazó, sztenderd eszközökkel feldolgozott és sztenderd formában, szabadon elérhető strukturált adatbázis létrehozását oroszországi kisebbségi uráli nyelvekre.

A projekt címe: *Az uráli nyelvek mondattanának változása aszimmetrikus kontaktushelyzetben*, időtartama másfél év (2016. február – 2017. július), befogadó intézménye az MTA Nyelvtudományi Intézete, projektvezetője É. Kiss Katalin. A projektet az NKFI támogatja, azonosítója: ERC_HU_15 118079. Ez egy pre-ERC projekt, amelynek az a célja, hogy lehetőséget adjon egy jövőbeli ERC[1] pályázat elméleti és módszertani alapjainak lefektetésére. A cikkben ismertetett elméleti és módszertani megfontolások a folyamatban levő pilot projekt során lettek kidolgozva, de természetesen a majdani ERC projektre is vonatkoznak.

A projektnek két fő célja van. Az elméleti cél egyrészt a kihalás szélén álló rokon nyelvek sajátos mondattani tulajdonságainak a leírása, másrészt ezen nyelvek szintaktikai változásainak vizsgálata, amelyek feltételezésünk szerint az orosz nyelv erőteljes hatására mennek végbe. A projekt másik célja egy annotált korpusz létrehozása *udmurt*, *tundrai nyenyec*, *szinjai* és *szurguti hanti* nyelvű, írott és beszélt nyelvi szövegekből, amely lehetővé teszi az uráli–orosz kontaktushatás kutatását. Ahhoz, hogy változásokat tudjunk detektálni, különböző korokból származó szövegeket kell gyűjtenünk és összehasonlítanunk. Az Oroszország területén beszélt kisebbségi uráli nyelvek esetében a legrégebbi írott nyelvi források a 19. század végéről származnak, amikor szervezett expedíciók keretében indultak terepmunkára etnográfusok, nyelvészek és egyéb szakemberek, hogy feltérképezzék a rokon nyelveket. Vagyis az általunk vizsgált régi szövegek a 19. század végéről – 20. század elejéről származnak. Emellett mai anyagot is gyűjtünk, nyomtatott és elektronikus forrásokból, illetve terepmunkán gyűjtött beszélt nyelvi adatokból.

A pilot projekt keretein belül mindegyik nyelvnek mindkét korából származó szövegeket gyűjtünk, és állítjuk elő legalább az eredeti szöveg kitisztított változatát. Az adatok minden szintű feldolgozását, IPA-átiratát, teljes morfológiai elemzését és legalább angol fordítását viszont csak kb. 4000 token/kor/nyelv mennyiségű adatra tervezzük a pilot projektben. Természetesen a majdani ERC projektben ennek sokszorosára lesz szükség ahhoz, hogy az egyes nyelvi jelenségek változásáról tényleges következtetéseket lehessen levonni.

A cikk további része az alábbiak szerint épül fel. A 2. fejezet a korpuszépítés gyakorlati lépései mögött meghúzódó elméleti és módszertani megfontolásokat mutatja be. A 3. fejezet ismerteti, hogy milyen szövegeket gyűjtöttünk és honnan, majd a 4. fejezet bemutatja az egyes szövegfeldolgozó lépéseket. Az 5. fejezet a korpusz felépítését írja le, és végül a 6. fejezet tartalmazza a konklúzióinkat és a jövőbeli terveinket.

[1] https://erc.europa.eu/

2. Elméleti megfontolások

A projekt nyelvei mind veszélyeztetettek és hiányosan dokumentáltak, de azért mutatkozik köztük némi különbség. Az udmurt nyelv több szempontból is kilóg a többi közül. Egyrészt Udmurtia egyik hivatalos nyelve, másrészt a nyelvi veszélyeztetettséget jelölő EGIDS-skálán [9,3] az 5., vagyis az *írott* kategóriába tartozik. Ez utóbbi annyit tesz, hogy a nyelvet napi szinten használják, és létezik egy sztenderd irodalmi változata, de az nem annyira terjedt el.

A projekt másik három nyelve mind szibériai nyelv, és mind a 6b, vagyis *veszélyeztetett* kategóriába tartoznak az EGIDS-skálán. Ezeket a nyelveket manapság már szinte csak az idősebb generáció használja, ők is csak családi és informális körben. Nem hivatalos nyelvek, továbbá alacsony presztízsűek, és a rájuk irányuló revitalizációs törekvések sem mondhatóak nagy számúnak és sikeresnek.

Ezek a tényezők több olyan következménnyel járnak, amelyeket figyelembe kell venni a korpuszépítés során, és amelyek a jól dokumentált, sok beszélős nyelvek esetében nem feltétlenül játszanak fontos szerepet.

A korpuszépítés során figyelembe vett egyik fő kritérium az volt, hogy – lehetőségeinkhez mérten – kövessük a nyelvi dokumentáció alapelveit. A nyelvi dokumentáció egy nyelv adatainak rögzítését, annotálását, megőrzését és disszeminációját jelenti, azaz gyűjtést, feldolgozást, annotációt, közzétételt, archiválást és tárolást [20]. Projektünkben a himmelmanni [6] értelemben vett elsődleges adatokat rögzítjük és dolgozzuk fel. Ezek olyan kommunikációs eseményekből származó nyelvi adatok, amelyek a hétköznapi nyelvhasználatot tükrözik, például dialógusok, elbeszélések, élettörténetek, vagyis nem irányított beszélgetések és nem feldolgozott szövegek, szólisták, kérdőívek.

A nyelvi dokumentáció súlypontjai az elmúlt évtizedekben áthelyeződtek (vö. [1,17]). A nyelvi dokumentáció új szemléletet és új eszközöket használ, a leírásban teljességre, egységességre és összehasonlíthatóságra törekszik. Ez utóbbiakra törekszünk mi is a korpuszépítés során, amelyek betartásához a számítógépes nyelvészeti eszközök és módszerek használata segítséget nyújt.

A teljességre törekvés azt jelenti, hogy abban a szellemben kell gyűjteni az anyagot, hogy az minél szélesebb körben használható legyen majd. Ezért az adatbázis-építés során arra törekszünk, hogy a lehető legtöbb szerzőtől válasszunk szöveget, és ezek minél több társadalmi osztályt, kort, nemet, dialektust és műfajt öleljenek fel. Továbbá az is fontos, hogy az eredeti felvétel, vagyis az audió- és/vagy videóanyag is elérhető legyen, hogy a leírások és következtetések ellenőrizhetők legyenek. Ahhoz, hogy az adatbázis tényleg hasznosítható legyen más területeken, így például szociolingvisztikai és antropológiai kutatásokhoz is, gazdagon kell metaadatolni minden nyelvi adatot.

Az egységesség és összehasonlíthatóság az adatbázis-építés minden szintjén megjelenik. Fontos egyrészt, hogy a nyelvi annotáció során nem követünk semmilyen nyelvészeti paradigmát, másrészt viszont szigorúan követünk bizonyos nemzetközi sztenderdeket, hogy a nyelvek és az eszközök közötti átjárhatóságot biztosítsuk.

A különböző nyelvű, különböző ábécét használó, különböző lejegyzést követő szövegek egységes reprezentációjához sztenderd Unicode-karaktereket használunk a teljes korpuszban (a projektben használt lejegyzési, átírási és írásrendszerekről részletesebben lásd a 4.1. fejezetet).

A hangok szintjén a Nemzetközi Fonetikai Ábécét (International Phonetic Alphabet, IPA) követjük. Erre azért van szükség, mert az uráli nyelvek lejegyzői hagyományosan a Setälä-féle [15] átírási rendszert használják (részletesebben lásd a 4.2. fejezetet), amely egyrészt nem egy egységes rendszer, másrészt nem ismert az uralisztikán kívül, ezért minden szövegnek automatikusan legeneráljuk az IPA-átiratát is.

A morfológia szintjén a lipcsei glosszázási szabályokat (Leipzig Glossing Rules, LGR)[2] követjük. A tokenek és a hozzájuk tartozó morfológiai információk egymásnak megfeleltetve, párhuzamosítva vannak megjelenítve. A glosszák az említett nyelvekre elérhető morfológiai elemzők kimenetéből állnak elő automatikus konvertálással (további részletekért lásd a 4.3. fejezetet), amiből az következik, hogy a morfológiai annotáció csak akkor lesz morféma szinten is megfeleltetve, ha az elemző képes szegmentálásra. Ebben az esetben, az LGR szabályait követve, kötőjellel választjuk el egymástól a morfémákat, illetve az őket jelölő kódokat. Az LGR tartalmaz egy ajánlott címkelistát is, amelyet követünk, de némileg kiegészítve, tekintve, hogy az eredeti lista nem fedi le az általunk elemzett nyelvek minden morfológiai jelenségét.

A nemzetközi szabványok követése az általunk alkalmazott formátumok terén is jelentkezik, ami jelen nyelvek esetében azért is fontos, mert minden nyelvi dokumentációs és nyelvfeldolgozó eszköz, amely ezekre elérhető, különböző ki- és bemeneti formalizmusokat követ, amelyek között a szabványos formátumok biztosítják az átjárhatóságot. Az általunk előállított összes szöveges állomány UTF-8 karakterkódolású sima szöveg fájl. A tokenszintű annotációk oszlopok formájában vannak reprezentálva sztenderd `tsv` fájlokban, amelyek bemenetül szolgálhatnak további nyelvfeldolgozó eszközök számára, vagy könnyen átalakíthatók `XML`-fájlokká.

3. Szöveggyűjtés

Ahogy fentebb említettük, arra törekszünk, hogy a korpusz reprezentatív mintája legyen az adott nyelvi közösség nyelvhasználatának. Ezt a törekvésünket azonban a 2. fejezetben kifejtett tényezők nagyban befolyásolják. Mivel a projektben vizsgált szibériai nyelvek esetében nemigen beszélhetünk sztenderd írásbeliségről, továbbá a nyelvet elsősorban az idősebb generáció használja, akik nem termelnek napi szinten elektronikus szöveges adatot, ezen nyelvek esetében nem támaszkodhatunk olyan, viszonylag könnyen elérhető forrásokra, mint a blogok, tweetek vagy a napi sajtó. Az is nehezíti továbbá a szövegek begyűjtését, hogy a korábbi, terepen gyűjtött anyagokat a kutatók jellemzően nem teszik publikussá. Ha mégis elérhető elektronikus formában valamilyen anyag, akkor az

[2] https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf

egyrészt nem túl sok, másrészt inkább eklektikus adathalmaz, mint szisztematikusan annotált korpusz.

Mindezekből következik, hogy a szöveggyűjtésnél eléggé meg van kötve a kezünk. A régi szövegek közé olyan folklór szövegeket válogattunk, amelyeket a 19. század végén – 20. század elején gyűjtöttek, és maga a terepen járt kutató adta közre a maga lejegyzési szisztémája alapján. A régi szinjai hanti szövegek Wolfgang Steinitz [16] gyűjtéséből származnak az 1930-as évekből, míg a szurguti hanti szövegeket Heikki Paasonen [18] gyűjtötte 1900-01-ben a Jugán folyó környékén. A régi udmurt szövegek két forrásból származnak: egyrészt Yrjö Wichmann [19] gyűjtéséből, ami 1901-ben lett publikálva, másrészt Munkácsi Bernát 1887-es terepmunkájából [10]. A régi tundrai nyenyec szövegek forrása Toivo Lehtisalo 1911–12-es gyűjtése [8]. Annak ellenére, hogy ezek mind folklór szövegek, vagyis ugyanabba a műfajba tartoznak, a szövegválogatást igyekeztünk úgy végezni, hogy a dialektusok és az adatközlők kora és neme szerint kiegyensúlyozott legyen. Az összes elérhető metaadatot összegyűjtjük, és táblázatba rendezve közreadjuk a projekt weboldalán.

Az új szövegek sokkal inkább különböző műfajú forrásokból származnak: az új hanti adatok lejegyzett interjúkat tartalmaznak, míg az udmurt szövegek a *Mynam malpanësy*[3] és a *Marajko*[4] nevű blogokból származnak. A modern tundrai nyenyec adat tartalmaz újságcikkeket a *Njar'jana Ngerm* című újságból, valamint új gyűjtésű folklór szövegeket Labanauskas [7] és Puškarëva–Chomič [14] gyűjtéseiből.

A beszélt nyelvi adatok a projektrésztvevők terepmunkái során gyűjtött és a jövőben gyűjtendő anyagaiból áll össze. Ezek a felvételek az ELAN-ban[5] lesznek lejegyezve és illesztve. Terveink szerint az új szövegek ugyanabból a régióból lesznek gyűjtve, ahonnan a régiek is származnak, hogy a nyelvjárási különbségeket kiküszöböljük a szintaktikai változások vizsgálata során.

4. Szövegfeldolgozás

A korpuszépítési workflow első lépése az eredeti szöveges anyag előállítása és egységes formátumra hozása, ezt írja le a 4.1. fejezet. A különféle lejegyzési és átírási rendszerek közötti átjárást biztosítanunk kell; az ehhez szükséges konverziós lépésekről a 4.2. fejezet tudósít. A korpusz morfológiai annotációt is tartalmaz, amelynek leírása a 4.3. fejezetben található.

4.1. Az eredeti szöveg előállítása

A *beszélt* nyelvi adatok feldolgozásának első lépése a lejegyzés, más néven transzkripció. Az uralisztikában a FUT (Finno-Ugric transcription) vagy más néven uráli fonetikai ábécé az elterjedt, amelyet Eemil Nestor Setälä [15] publikált 1901-ben azzal a szándékkal, hogy az uralisták által használt lejegyzési rendszereket

[3] http://udmurto4ka.blogspot.hu/
[4] http://marjamoll.blogspot.hu/
[5] http://tla.mpi.nl/tools/tla-tools/elan/

egységesítse. Ennek ellenére a FUT-ba sorolt lejegyzések nem alkotnak egy következetes rendszert, sőt igen jellemző, hogy ugyanannak a hangnak a jelölésére más és más karaktert használnak.

Miután megtörtént a beszélt nyelvváltozat lejegyzése, az adat onnantól kezdve ugyanazokon a feldolgozási lépéseken megy keresztül, mint az írott nyelvi anyag. A régen lejegyzett és kiadott szövegek is lejegyzett beszélt nyelvi anyagnak számítanak a további feldolgozás szempontjából.

Az általunk feldolgozni kívánt *írott* nyelvi adatok egy része csak nyomtatott könyv formájában volt elérhető, ezért ezeket beszkenneltük, majd optikai karakterfelismerő (OCR) program segítségével jutottunk hozzá a szöveghez. A korpuszunkban található nagyszámú lejegyzési és írásrendszer kezelése miatt az OCR programmal szemben alapvető elvárásunk volt a taníthatóság. Az Abbyy FineReader Professional Edition[6] mellett döntöttünk, ami ugyan nem nyílt forráskódú, de meglehetősen könnyen tanítható, és elég jó minőségű kimenetet ad.

Bizonyos dokumentumokat a webről töltöttünk le; ebben az esetben `HTML`-forrásokból és `PDF`-fájlokból kellett kinyernünk a szöveget. A kimenetet minden esetben kézzel ellenőriztük, hogy a következő feldolgozó lépésben minél tisztább anyaggal dolgozhassunk.

A szabványosság előnyei miatt a teljes korpuszt sztenderd UTF-8 kódolású Unicode-karakterekkel tároljuk és jelenítjük meg. Mindenképpen szükséges egy az egész korpuszra kiterjedő szigorúan egységes formátum, ez teszi lehetővé, hogy a lekérdezéseket az egész anyagra vonatkoztathassuk. Ezt csak úgy biztosíthatjuk, ha következetesen betartjuk azt az alapelvet, hogy azonos dolgokat mindig ugyanúgy, különbözőeket pedig mindig eltérően jelölünk.

Ennek eléréséhez az első lépés az volt, hogy létrehoztunk egy egységes karaktertáblát, amelyben minden nyelv minden transzkripciós, transzliterációs és írásrendszerének minden karaktere szerepel a Unicode-kódjával és -nevével, valamint Prószéky-kódjával egyetemben. Ez a kódtábla van használva minden szövegfeldolgozó lépésnél: ezekkel a karakterekkel történik a hangzó szövegek lejegyzése, ezekre a karakterekre tanítjuk be az optikai karakterfelismerőt, ezekre a karakterekre normalizáljuk a különböző forrásokból származó szövegeket, és ezek szolgáltatják a különböző irányú konverziók bemeneti és kimeneti karakterállományát is (lásd a 4.2. fejezetet).

A következő lépésben ellenőrizzük és normalizáljuk az összes szöveget egy `Perl`-szkript[7] segítségével, amely kilistázza a dokumentumban szereplő Unicode-karaktereket. A lista alapján könnyedén felismerhetők és eltávolíthatók az idegen nyelvű részek, illetve a nem helyesen használt karakterek lecserélhetők.

4.2. Átírás és konverzió

A transzkripcióval szemben meg kell különböztetnünk a transzliterációt, amely egy már írott formában létező nyelvi adat átírása egy másik írás- vagy jelölési

[6] http://finereader.abbyy.com/
[7] https://gist.github.com/takdavid/3fa2cc3ae21aa96da24b8bd90b8c63b0

rendszerre. Ahogy említettük, az adatbázisunk tartalmaz minden szöveget legalább az eredeti lejegyzésében, amelyet a nyelv dokumentálója használ, valamint IPA-átírásban is. Ez utóbbit azért tartjuk fontosnak, mert így nem csak az uralisztika kutatói, hanem más nyelvészek is olvasni és használni tudják az anyagot. Továbbá – mivel az érintett nyelvek írásrendszere a cirill ábécén alapszik – megőrizzük az eredeti cirill írást, amennyiben van ilyen. Ha nincs, de szükség van rá a morfológiai elemzőhöz, akkor egy konverziós lépés során előállítjuk. Ugyanígy járunk el a különféle FUT-típusú lejegyzésekkel is: mivel bizonyos morfológiai elemzők csak bizonyos módon lejegyzett szövegeket fogadnak el inputként, ezeket is elő kell állítani egy konverziós lépés során. (A morfológiai elemzőkről lásd a 4.3. fejezetet.)

A projektben vizsgált négy nyelvre összesen 11 konverziós irány van, amelyekre konvertereket fejlesztettünk. A régi szinjai hanti szövegek eredetileg Steinitz lejegyzésével készültek, aki a saját FUT-jellegű rendszerét használta. Ezt konvertáljuk először IPA-ra, aztán arra a szintén FUT-jellegű ábécére, amelyet az általunk használt morfológiai elemző fejlesztői alkalmaztak. Az új szinjai hanti szövegek lejegyzése már eleve ez utóbbi szerint zajlik.

A régi szurguti hanti szövegeket az Ob-Ugric Database (OUDB)[8] fejlesztői bocsátották a rendelkezésünkre, és mivel ők csak IPA-ban tették elérhetővé az anyagukat, nekünk is csak IPA-átiratunk van. A modern szurguti hanti szövegek viszont a mai cirill betűs hanti írással íródtak, amelyet először átkonvertálunk a Csepregi Márta [2] által alkotott és a hanti nyelvet kutatók körében széles körben használt átírásra, majd ebből állítjuk elő az IPA-verziót.

Az udmurt nyelv esetében négy különböző konverterre van szükség. Először létrehoztuk a konverziós szabályokat a Munkácsi–IPA és a Wichmann–IPA irányokba, majd az IPA-verziót konvertáljuk cirill betűs írásmódra. Ez utóbbira azért van szükség, mert az udmurtra fejlesztett morfológiai elemzők mindegyike cirill betűs bemenetet vár. Az új udmurt szövegek esetében az irány fordított, vagyis a cirill szöveget konvertáljuk IPA-ra.

A régi tundrai nyenyec szövegek bizonyos értelemben kivételt képeznek. Lehtisalo olyan bonyolult transzkripciós rendszert dolgozott ki, amelyre se az IPA-átírás elkészítéséhez, se a morfológiai elemzőhöz nincs szükség, továbbá egy részük nem is lenne reprezentálható sztenderd Unicode-karakterekkel. Ezért a Lehtisalo-szövegek OCR-ezésénél egy Lehtisalo–Hajdú leképezést használtunk, így ezek a szövegek már eleve Hajdú Péter [5] transzkripciója alapján készültek el. Ez utóbbi lett IPA-ra, majd cirillre konvertálva, az utóbbi a morfológiai elemzőhöz. A modern nyenyec szövegekkel hasonló a helyzet, mint az udmurttal: a cirill betűs modern nyenyec írásnak is elkészítjük az IPA-konverzióját.

A konverzió első lépéseként az adott nyelv szakértői átírási szabályokat definiáltak. Ezek lettek kiterjesztett reguláris kifejezéseket tartalmazó helyettesítési parancsokká átalakítva, és így beadva a `sed` parancsnak segédfájlként egy `-f` kapcsolóval. Vagyis ez egy szabályalapú rendszer, annak minden tipikus előnyével és hátrányával. Hátrányai közé tartozik, hogy nyelvfüggő, sőt jelen esetben irányfüggő, vagyis nem vihető át egy másik konverziós irányra változtatás nélkül.

[8] http://www.oudb.gwi.uni-muenchen.de/

Ezen kívül, ha sok szabállyal dolgozunk, amelyeknek fontos a sorrendje is, nem mindig egyszerű fejben tartani az összeset, így könnyű hibázni, ami tökéletesen rossz eredményhez vezethet. Van viszont egy nagy előnye a szabályalapú rendszereknek, mégpedig az, hogy magas pontosságot produkálnak. Mivel az automatikusan konvertált szövegeket nyelvész szakértők ellenőrzik a projektünkben, mi a magas pontosság mellett voksoltunk, a fent említett hátrányok ellenére is.

4.3. Morfológiai elemzés

A korpusz egy része morfológiai szintű annotációt is tartalmaz. Ezekben a szövegmintákban minden tokennél megadjuk a lemmát, a szófaji címkét és az angol glosszát. Ezek az információk a rendelkezésre álló morfológiai elemzők kimeneteiből lesznek konvertálva. Ehhez első lépésben meg kell csinálni egy leképezést, amely a különböző morfológiai elemzők által használt címkekészletet képezi le az általunk létrehozott egységes morfológiai címkekészletre. Ez utóbbiban és a glosszázás során általában is az LGR konvencióit és rövidítéseit követjük, kisebb kiegészítésekkel.

Az általunk vizsgált négy nyelvből háromra létezik morfológiai elemző, amelyet tudunk használni a morfológiai annotáció előállításának nyelvtechnológiai támogatására. Ennek ellenére az annotáció nem teljesen automatikusan készül, hanem kézi javítást is igényel.

A legismertebb szövegfeldolgozó keretrendszer kis uráli nyelvek nyelvtechnológiai támogatására a Giellatekno[9], amelynek keretein belül mások mellett helyesírás-ellenőrzők, digitális szótárak és morfológiai elemzők is fejleszthetők. Ez utóbbi már létezik, bár folyamatosan fejlesztés alatt áll, az udmurt, az északi hanti és a tundrai nyenyec nyelvekre (az északi hantinak egy aldialektusa a szinjai hanti).

Emellett létezik egy másik morfológiaielemző-csomag is kis uráli nyelvekre, így udmurtra és szinjai hantira, a MorphoLogic Kft. és az MTA Nyelvtudományi Intézetének közös munkájának eredményeként [12,4]. Ezek az elemzők nem szabad forráskódúak, hanem egy online felületen keresztül érhetők el[10]. A kimenetük egy `HTML`-fájl, amely minden beadott token minden lehetséges elemzését tartalmazza. A kézi munka megkönnyítéséhez egy webes felületet használunk, amely eredetileg ómagyar szövegek morfológiai egyértelműsítéséhez lett kifejlesztve [13], de némi módosítással a mi céljainkra is használható. A felhasználó az egyértelműsítendő token fölé egerészik, majd az összes elemzést tartalmazó legördülő menüből kiválasztja a helyes elemzést. Azokhoz a szavakhoz, amelyeket nem ismert fel az elemző, kézzel kell bevinni a helyes elemzést. Ez a webes interfész a Giellatekno outputján is használható.

A szinjai hanti és az udmurt szövegek elemzésére a morphologicos elemzőt használjuk, mert ez morféma szinten szegmentált kimenetet ad, továbbá a magyar (és a szinjai hanti esetében az angol) fordítást is előállítja.

[9] http://giellatekno.uit.no/
[10] http://www.morphologic.hu/urali/

A tundrai nyenyec szövegek elemzésére a Giellatekno elemzőjét használjuk. Mivel az elemző szótára a tundrai nyenyecnek csak egy dialektusába tartozó szavakat tartalmazza, valamint a nyelvtanfájlok egy korábbi nyelvtan alapján készültek, terveink között szerepel egyrészt a szótár bővítése egyéb nyelvjárásokba tartozó elemekkel, másrészt a nyelvtanfájlok update-elése a legújabb nyelvtan [11] alapján.

Sajnos a negyedik nyelvre, a szurguti hantira nem tudunk elérhető morfológiai elemzőről, de azért megpróbáltunk erre a nyelvre is valamilyen automatikus támogatást nyújtani. Amit alkalmaztunk, az egy végtelenül egyszerű memóriaalapú megoldás. Zipf törvénye alapján tudjuk, hogy a néhány leggyakoribb szó lefedi a teljes szöveg nagy százalékát. Ebből kiindulva kilistáztuk a modern szurguti hanti szöveg minden olyan tokenjét, amely legalább ötször előfordul. Ezekhez egy nyelvész szakértő kézzel hozzárendelte a szófaji kódot, az inflexiós címkéket és a lemma angol fordítását. Ezzel a glosszák több mint 60%-át tudjuk automatikusan generálni, ami nagy mértékben csökkenti a kézi munka mennyiségét.

5. A korpusz felépítése

A korpusznak három fő annotációs szintje van. A transzkripció és a transzliteráció, vagyis az eredeti szöveg és az átírások szintje, a morfológiai elemzés szintje, valamint a fordítások szintje. Minden dokumentumhoz minden szinten legalább egy szövegverziónak meg kell lennie. Ezek a kötelező verziók sorrendben a következők: az IPA-átírás, a lemma, a szófajcímke és az angol glossza, valamint az angol fordítás. Az átírások és a morfológiai elemzés szintjén az annotáció tokenszintű, vagyis minden egyes tokenhez megadjuk legalább az IPA-átiratát és az előbb felsorolt morfológiai információkat. A fordítás ezzel szemben mondatszintű annotáció, vagyis teljes mondatokhoz rendelünk legalább angol, de sokszor magyar, német és orosz fordítást is. Ez utóbbiak teljes mértékben kézzel készülnek.

A token- és mondatszintű annotációkat tartalmazó szövegfájlokat beimportáljuk az ELAN-ba, ahol mondat szinten időben illesztve lesznek az audió- vagy videóanyaghoz. Az ELAN az annotációs szinteket horizontális szintekként jeleníti meg, amit az 1. táblázat illusztrál egy tundrai nyenyec példával.

6. Konklúzió és jövőbeli kutatási irányok

Cikkünkben bemutattunk egy pilot projektet, amely azt tűzte ki célul, hogy annotált nyelvi adatbázist épít négy oroszországi kisebbségi uráli nyelvre. A célkitűzést az indokolja, hogy ezeken a nyelveken jól vizsgálható az orosz–uráli kontaktushatás, amely a projekt elméleti célja, valamint hogy az uralisztika területén inkább eklektikus adathalmazokkal találkozik a kutató, mint szisztematikusan annotált adatbázisokkal. Meggyőződésünk, hogy a számítógépes

1. táblázat. Token és mondat szinten illesztett tundrai nyenyec szöveg.

YRK Hajdú:	jā	mī̆daxana	amkerta	jaŋkūwi
YRK IPA:	ja	mi:daxana	ămkerta	jăŋkuwi
YRK Cyrillic:	я	мыдахана	амкэрта	яӈкувы
lemma:	я	мы	ӈамгэ	яӈгось
POS:	N	Ptcp	Pron.neg	V
glossza:	earth	create.IPFV.PTCP.LOC	nothing	neg.EX.INFER.3SG
ENG:	when the earth was created, there was nothing			
GER:	zur zeit der erschaffung der erde gab es nichts			
HUN:	a Föld teremtésének idején nem volt semmi			

nyelvészet eszköztára jól használható az ilyen speciális nyelvekre történő korpuszépítés során is, és nagyban segíti az uralisták és az elméleti nyelvészek munkáját.

A cikkben leírt elméleti és módszertani megfontolások nem csak a pilot projektben, hanem a majdani ERC-projektben is hasznosíthatóak lesznek, míg a pilot projekt során épített korpusz anyaga a jövőben bővítésre szorul.

A korpuszépítés során követjük az open access filozófiáját, amelynek két vetülete is van. Egyik, hogy törekszünk arra, hogy szabadon elérhető eszközöket használjunk, valamint hogy újrahasznosítsunk már valamilyen formában publikált adatokat is. Másrészt a projekt eredményeképpen előálló minden szöveges és feldolgozó erőforrást szabadon hozzáférhetővé teszünk a projekt weboldalán: http://www.nytud.hu/oszt/elmnyelv/urali/adatbazisok.html.

Távolabbi terveink között szerepel, hogy az adatbázis ne csak letölthető formában legyen elérhető, hanem egy online lekérdező felületen keresztül is, amely a számítógépes eszközök használatában kevésbé jártas kutatók számára is lehetőséget nyújt az adatok használatára. Ezenfelül, a hosszú távú megőrzés jegyében, az általunk létrehozott összes adatot szeretnénk elérhetővé tenni egy nemzetközi nyelvi archívumon keresztül is, mint amilyen a The Language Archive[11] által működtetett DOBES (Documentation of Endangered Languages) korpusz.

7. Köszönetnyilvánítás

A projektet az NKFI támogatja, a pályázat azonosítója: ERC_HU_15 118079.

Az elméleti alapok lefektetésében és a korpuszépítésben több kutató is részt vett; a korpusz nélkülük nem jött volna létre. Ők név szerint: Asztalos Erika, Gugán Katalin, Kalivoda Ágnes, Mus Nikolett, Nguyen-Dang Nóra Lien, Ruttkay-Miklián Eszter, Tánczos Orsolya.

Külön köszönettel tartozunk az OUDB projekt vezetőjének, Elena Skribniknek, hogy rendelkezésünkre bocsátotta a Paasonen-szövegeket; Schön Zsófiának,

[11] https://tla.mpi.nl/

hogy rendkívül sokat segített a szurguti hanti szövegek IPA-átírásával kapcsolatban; A. S. Pesikovának és A. N. Volkovának, hogy engedélyezték nekünk az általuk felvett és lejegyzett interjúk felhasználását.

Hivatkozások

1. Blokland, R., Fedina, M., Gerstenberger, C., Partanen, N., Rießler, M., Wilbur, J.: Language documentation meets language technology. In: First International Workshop on Computational Linguistics for Uralic Languages. pp. 8–18. No. 2 in Septentrio Conference Series (2015)
2. Csepregi, M.: Szurguti osztják chrestomathia. Szeged (2011)
3. Fazakas, N.: Újabb fejlemények a nyelvi revitalizáció kutatásában. Nyelv- és irodalomtudományi közlemények LVIII.(2), 155–164 (2014)
4. Fejes, L., Novák, A.: Obi-ugor morfológiai elemzők és korpuszok. In: VII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2010). pp. 284–291. Szegedi Tudományegyetem (2010)
5. Hajdú, P.: Chrestomathia Samoiedica. Tankönyvkiadó, Budapest (1989)
6. Himmelmann, N.P.: Linguistic data types and the interface between language documentation and description. Language Documentation and Conservation 6, 187–207 (2012)
7. Labanauskas, K.I.: Neneckij fol'klor. Mify, skazki, istoričeskie predanija. Vyl. 5. Krasnojarsk (1995)
8. Lehtisalo, T.: Juraksamojedische Volksdichtung. Suomalais-Ugrilainen Seura, Helsinki (1947)
9. Lewis, M.P., Simons, G.F.: Assessing endangerment: Expanding Fishman's GIDS. Revue Roumaine de Linguistique 55(2), 103–120 (2010)
10. Munkácsi, B.: Votják népköltészeti hagyományok. Magyar Tudományos Akadémia, Budapest (1887)
11. Nikolaeva, I.: A Grammar of Tundra Nenets. Mouton de Gruyter (2014)
12. Novák, A.: Morphological Tools for Six Small Uralic Languages. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06). pp. 925–930. ELRA (2006)
13. Novák, A., Orosz, G., Wenszky, N.: Morphological annotation of Old and Middle Hungarian corpora. In: Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. pp. 43–48. Association for Computational Linguistics, Sofia, Bulgaria (August 2013), http://www.aclweb.org/anthology/W13-2706
14. Puškarëva, J.T., Chomič, L.V.: Fol'klor nencev. Novosibirsk (2001)
15. Setälä, E.N.: Über Transskription der finnisch-ugrischen Sprachen. Finnisch-ugrische Forschungen 1, 15–52 (1901)
16. Steinitz, W.: Ostjakologische Arbeiten. Akadémiai Kiadó, Budapest (1975)
17. Szeverényi, S.: Rendkívül rövid bevezetés a dokumentációs nyelvészetbe. In: Szeverényi, S., Szécsényi, T. (eds.) Érdekes nyelvészet, pp. 146–157. JATE Press, Szeged (2015)
18. Vértes, E. (ed.): Heikki Paasonens surgutostjakische Textsammlungen am Jugan. Neu transkribiert, bearbeitet, übersetzt und mit Kommentaren versehen von Edith Vértes, Mémoires de la Société Finno-Ougrienne, vol. 240. Suomalais-Ugrilainen Seura, Helsinki (2001)

19. Wichmann, Y.: Wotjakische Sprachproben II. Sprichwörter, Rätsel, Märchen, Sagen und Erzählungen. Helsinki (1901)
20. Woodbury, A.C.: Language documentation. In: Austin, Peter K.; Sallabank, J. (ed.) The Cambridge Handbook of Endangered Languages, pp. 159–186. Cambridge University Press (2011)

First Experiments and Results in English–Hungarian Neural Machine Translation

László Tihanyi, Csaba Oravecz

I.R.I.S. / European Commission, Directorate-General for Translation
e-mail:{laszlo.tihanyi,csaba.oravecz}@ext.ec.europa.edu

Abstract. Neural machine translation (NMT) has emerged recently as a promising alternative of standard rule-based or phrase-based statistical approaches especially for languages which have so far been considered challenging for the two paradigms. Since Hungarian has long been one of these challenging languages, it is a natural candidate for neural machine translation to explore whether this approach can bring some improvement in a task which translation models have so far been unable to cope with. The paper presents our first results of applying neural models to English to Hungarian translation and shows that with the right configuration and data preparation, publicly available NMT implementations can significantly outperform the previous state-of-the-art systems on standard benchmarks.

Keywords: neural machine translation, attention based model, source side linguistic analysis, morphology-aware subword units, alignment-based unknown word replacement

1 Introduction

Neural machine translation is a relatively young but already very successful approach in one of the most difficult areas of natural language processing. Recent results, in particular those at WMT'16 [1] have made the attention of the machine translation community shift considerably towards neural networks, and it seems more and more plausible that the field is experiencing a paradigm shift. After the early rule-based approaches phrase-based SMT has been dominant for many years but now the new models promise progress even for languages deemed difficult for standard translation systems. Although some of the main drawbacks of current neural systems are already widely known, such as the occasional lack of adequacy, NMT has been reported to work well with rich morphology and significant word reordering, producing more fluent output [2,3]. This motivates the present work in an attempt to create an English to Hungarian end-to-end neural translation system.

The aim of this paper is not so much to give a detailed account of all the system components and provide an extensive description of the many experiments but rather to present a general overview with focus on the best setups and most

important findings. We compare the performance of a Moses based [4] in house translation architecture (MT@EC [5]) with neural systems based on two publicly available implementations and report results that even in this early stage convincingly show that NMT is not only comparable but promisingly better on the EU domain than earlier phrase-based statistical or rule-based approaches [6] for the En-Hu direction.

2 Neural Machine Translation

Standard phrase-based statistical machine translation is built around a number of components, while a(n attention based) NMT system is in principle an end-to-end encoder-decoder framework to model the entire translation process [7]. The role of the encoder, which is often implemented as a bidirectional recurrent network, is to summarise the source sentence into a set of context vectors, and the decoder acts as a recurrent language model to generate a target sentence word after word by leveraging information from the weighted sum of the context vectors at each step, where the weights are computed by the attention mechanism [8]. In learning, the whole model is jointly trained to maximise the conditional probability of a gold standard translation given a source segment from a training corpus of parallel sentences. Learning is regulated by various optimisation algorithms with backpropagation.

3 Previous Work

The evolution of neural translation systems has been rapid. In the beginning, neural networks were only used as a component in a classical SMT system [9,10]. But soon after the success of the end-to-end neural network based translation system from Montreal University at WMT15 [11], more and more work has been invested to develop direct neural translation models for several language pairs, predominantly for Indo-European languages and Chinese [12,13] but some early results have already been published on for example Arabic as well [14].

At the latest WMT conference, the Edinburgh Neural Machine Translation System was clearly dominant [15], and since then NMT has already found its way even into production systems [16,17]. The field is still evolving swiftly and many techniques have been and are being developed to further improve performance, some of them alrady being close to standard, such as the attention mechanism or using subword units to overcome the unknown/rare word problem [18]. Further attempts have been made to change the basic input (output) unit from the wordform to the character [19], to better model translation coverage [20], or to use a more suitable optimisation technique, which instead of maximising the likelihood of the training data, can take evaluation metrics as loss functions and aims to minimise expected loss on the training data [21]. Many more new directions are investigated and there is constant progress with the promise of offering new solutions to old problems of MT.

4 System Architecture

Our pilot En-Hu system consists of five basic building blocks (steps). The core component is the translation module (see Section 4.4), which can be of different types and used more or less interchangeably within a pre- and post-processing pipeline. This, however, must eventually be tailored a bit to each of the core translation modules. In our full fledged neural architecture the components are the following: i) pre-processing pipeline, ii) configuration frontend to the NMT toolkits, iii) core NMT module iv) post-processing module, v) evaluation and visualisation module.

4.1 Experiment management

Finding the optimal settings for all (hyper-)parameters for an NMT system requires a large number of experiments with a wide scale of different values. In the NMT toolkit implementations this is not (yet) conveniently supported. Therefore we have developed a simple Python frontend which supports the setting of all relevant parameter values in one unified configuration file, which is used by the core neural training, the pre- and post-processing and the test process at the same time. This not only makes running experiments with a range of parameter values simple but also keeps a record of all the settings that define a specific translation model.

The whole system is managed by two wrapper shell scripts that take the configuration file and run the processing steps with the parameter settings as defined therein. One script is responsible for pre-processing and training while the other for translation (including pre-processing), post-processing and evaluation.

4.2 Solving the rare/unknown word problem

For neural translation models two types of approaches have been proposed to alleviate the problem that only a limited (50-100k word) vocabulary is allowed for the model to remain computationally manageable. The first branch is based on extending the base encoder-decoder model to incorporate external information, such as alignment information and dictionary look-up for out-of-dictionary words [22], the second tries to transform the input data into units smaller than a full word form, thereby constraining the number of possible forms [18]. Our solution attempts to utilise the advantages of both approaches.

On the Hungarian target side we apply morphological analysis [23] to facilitate morphology-aware tokenisation and split the target word forms into linguistically motivated stems and suffixes, [1] and so reduce the number of possible units. Further reduction is achieved by an extensive placeholder mechanism tailored for each language pair. This mechanism is different from the one used in the in house MT@EC translation system. In MT@EC, placeholder replacement

[1] It is possible to further process this format with the algorithm of [18], but we do not yet have results for this type of setup.

is based upon hard alignment information, which is extractable from a standard phrase-based (Moses) system. In NMT, however, this information is not available, and so placeholder replacement uses segment level unique identifiers and language specific mapping tables to find the appropriate target forms.

For the remaining unknown words external information is utilised in the form of an alignment dictionary generated by a pre-processing step on the parallel data using the tool `fast_align` [24]. The source side equivalents of target side unknown tokens are identified from the "soft" alignment[2] based on the attention weights, potentially constrained by the guided alignment strategy of [25].

4.3 Pre- and Post-processing

Most of the pre-processing steps contain standard transformations on the data, such as tokenisation with the default Moses tokeniser, some normalisation, segment filtering by length, truecasing and placeholder replacement. It is possible to filter out training segments according to the ratio of unknown words but we have not experienced significant differences in performance by adding this filter to the pre-processing pipeline. For the unknown word replacement, the alignment dictionary is generated after the target side morphological split to find better translation equivalents between English and Hungarian input units. To support the processing of full documents, sentence segmentation is also implemented in the workflow.

In neural models it is very easy to support the incorporation of source side linguistic analysis into the model [26] by simply concatenating the additional information with the input token representation, creating a "neural equivalent" of a standard phrase-based factored model. We use the Wapiti toolkit [27] to provide part-of-speech and chunk labels on the English source side for some of the experiments (Figure 1). Although performance improvements have been reported using the output of higher level of linguistic analysis (full parsing) [26], in a future production environment processing time is critical and full parsing being prohibitively slow we try to use tools which are sufficiently fast not to delay the translation time significantly.

raw	OJ L 302, 19.10.1989, p. 1. Directive as last amended by Commission Decision 2006/60/EC (OJ L 31, 3.2.2006, p. 24).
pp	`_oj_num-1|NN ,|PUNCT _num_datum-1|NN ,|PUNCT p.|NN _num-1|CD .|PUNCT Directive|NNP as|IN last|JJ amended|VBN by|IN Commission|NNP Decision|NNP _num_inst-1|NN (|-LRB- _ojpage-1|CD )|-RRB- .|PUNCT`

Fig. 1. Raw and pre-processed source segment.

The translation output is post-processed to replace unknown tokens using the soft alignment information and the dictionary, and to insert the target surface forms for the the placeholders. If target side splitting is switched on then

[2] Which is in effect a probability distribution over possible tokens.

target surface forms are generated from stems and suffixes. In the end, the standard steps of segment recasing and detokenisation restore the final output of the system, which can be evaluated with BLEU and sentence level METEOR scores. For detailed investigation of the translation process, the visualisation of soft alignments is also implemented (Figure 2). In document translation mode translation memories in TMX format are generated as output.

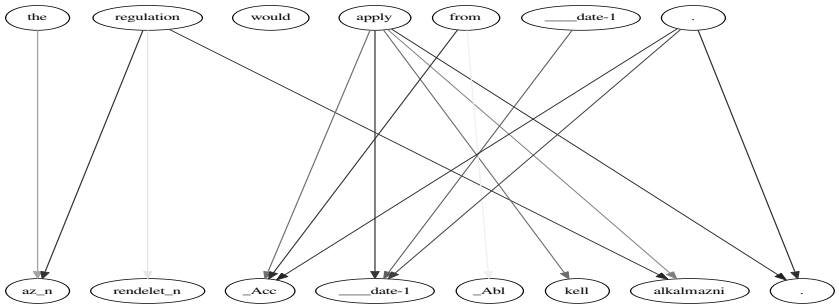


Fig. 2. Soft alignment between source and morpheme split target.

4.4 MT Systems

The baseline system is a Moses-based framework that is currently used in the MT@EC service at the European Commission. It does not use the most recent Moses version but in its current setup newer Moses versions give no increase in translation quality.[3] There have been many in house experiments to improve the En-Hu system in recent years but no significant progress has been made so we use a fairly basic phrase-based architecture in lack of any better alternative as our baseline system.

The neural systems are built around two publicly available implementations. The first one is called Nematus[4] [15], originally forked from Kyunghyun Cho's DL4MT tutorial repository[5], and is a Theano based toolkit using gated recurrent units (GRU) [28], which supports the output of alignment information, right-to-left rescoring, subword units and source side linguistic factors. The other is a Torch implementation[6] of an LSTM [29] based network architecture, maintained by the HarvardNLP group and SYSTRAN [16] and so is already used

[3] The novelty of recent versions is mostly in the support of advanced, more complex models, which on the one hand cannot be efficiently used in a production environment and on the other hand do not work better for the En-Hu language pair.

[4] `https://github.com/rsennrich/nematus`

[5] `https://github.com/nyu-dl/dl4mt-tutorial`

[6] `https://github.com/harvardnlp/seq2seq-attn`

in a production environment. This toolkit offers a wide range of parameter settings and configurations including the use of character level models or external embeddings, supports linguistic input features as well and comes with a built-in mechanism for alignment-based unknown word replacement using an external dictionary.

5 Experiments

5.1 Datasets

To find the best settings and values for the set of parameters that most influence translation quality it is necessary to run a large numbers of experiments, each taking significant time to run through. For this and other circumstantial reasons the initial experiments are run on the DGT-TM public dataset [30], originally with 2 million segments used for training, 3000 for validation and 1000 for testing. It has turned out, however, that the datasets contains a lot of duplicate segments and filtering them out not only shortens experiments, makes the model more compact but most importantly has no negative impact on translation qualittranslation quality. In fact, using only the filtered 1.1 million training set improved the Nematus-based system considerably (see Section 6).

5.2 Hardware environment

Training neural models requires substantial resources with dedicated hardware. Our models have been trained on two different architectures, one with NVIDIA GeForce GTX 980 GPUs with 4GB RAM, which allowed for one single training per GPU with some of the settings limited to not necessarily the optimal values, such as minibatch size or vocabulary, although our initial experiments have so far shown that for this magnitude of training data a larger vocabulary (100k word) does not lead to performance improvement. The other environment is provided in the IBM cloud by the CEF eTranslation project and contains 4 powerful servers with dual core Tesla K80 (2x12GB RAM) GPUs, which allow for at least two trainings and further two test runs in parallel. Training time is not significantly different on the two architectures, it takes around 2 days to reach optimal performance on these medium size datasets. Decoding time is also manageable, with less than one second per segment using the GPU.

5.3 Training Procedure

During training the optimisation algorithm and related hyper-parameters are pivotal to reach the best performance of the model. Similarly to the Google NMT research group [17] in our best systems (so far) we use a combination of the Adam gradient descent optimisation algorithm [31] and Stochastic gradient descent (SGD) but the exact values always depend on the particular dataset and language pair, hence there are no globally valid settings for all scenarios.

On average, Adam quickly converges at around 150k updates when we switch to SGD starting with a learning rate of 0.1, which we anneal gradually by a magnitude after a certain number of steps, depending on the actual dataset. On average, in our case this process could lead to 1 point increase in the BLEU score.

We have not experimented with dropout[7] to avoid overfitting. Dropout is normally used for limited datasets and we expect it will not be necessary once full size parallel data can be fed to the systems, which in general contain well more than 10M segments for most of the EU language pairs.

Ensembles of neural models have been shown to perform better than a single model[8], however, we have not extensively tested them so far. Running an ensemble always requires more resources, which can be a problem in our production environment so if ensembling is not absolutely necessary a single model can still be a good compromise.

Table 1. BLEU scores for various En-Hu MT systems.

	Model	Abbreviation	BLEU
0	Baseline phrase-based	SMT	23.37
1	First basic neural model	NMT-BASE	22.05
2	Neural model with target side split	NMT-MORPH	21.94
3	Increased segment length (75)	NMT-LENGTH	23.02
4	Hyphenated compound split	NMT-SPLIT	24.58
5	Unique training segments	NMT-UNIQE	26.11
6	Subword units	NMT-BPE	25.95
7	Source side factors	NMT-FACTOR	26.13
8	HarvardNLP toolkit base model	HNLP	24.90

6 Results

We have run a large number of experiments with many settings and are still in search of the best models hoping to increase the translation quality further with better pre-processing or more optimal parameter values. In Table 1 we summarise the best results of selected models while Figure 3 illustrates how model performance improves during training. In most of the models, we use a 50k word vocabulary, and a minibatch size of 32.

We note that initially we experimented with a maximal segment length of 50-60, inherited from the phrase-based system. Using a higher value (75-100) significantly increased the performance of the NMT systems (Model 3) but interestingly had only negligible effect on SMT. The same tendency seems to be

[7] Randomly dropping units from the network during training.
[8] See eg. [15] but this technique is ubiquitous in all NMT systems.

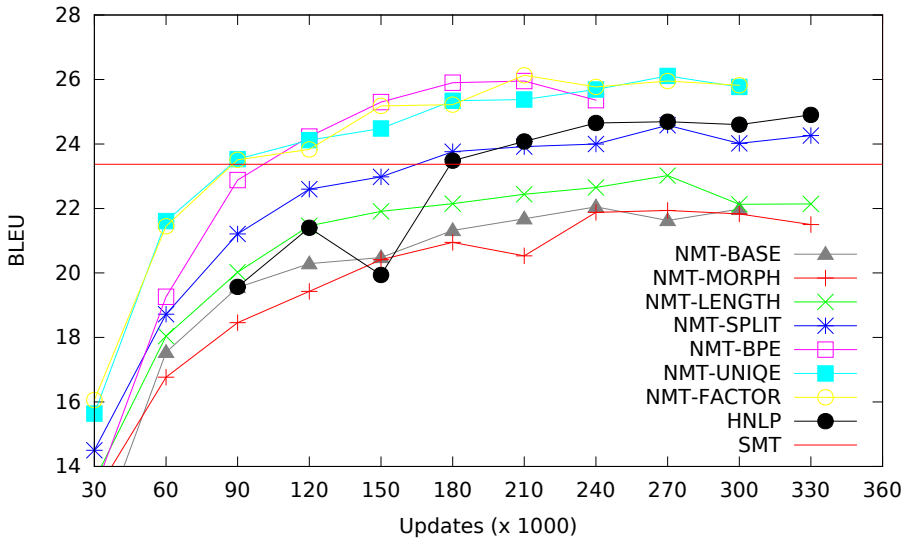


Fig. 3. BLEU score vs. number of updates during training.

true for the use of source side factors (Model 7), splitting the hyphenated compunds on both source and target side (Model 4) or duplicate segment filtering (Model 5): they are useful for NMT but almost totally useless for the Moses-based system (therefore we do not separately report these scores for the SMT system). Morphology aware tokenization in itself does not give improvement (Model 1 vs. Model 2) but our current implementation is extremely rudimentary without any disambiguation and with many errors in generation. We expect much better results from the new processing tools for Hungarian that we are now building into our system.

Our best model achieves a BLEU score of 26.13 which is about 3 points higher than the phrase-based baseline. In Figure 4 we illustrate this difference with sample translations from selected engines together with the reference translation and the segment level METEOR scores. On a casual inspection the good quality (and fluency) of the neural translation is striking, and (informal) manual evaluation tends to prefer the neural translation even in cases where the automatic score favours the phrase-based alternative. In this stage of the project, there has not yet been rigorouos qualitative evaluation but some early feedback from professional translators seems to confirm much of the findings of [3]: the NMT output has better morphology, word order and agreement, and post editing effort is estimated much lower. It is true, however, that without a large scale manual evaluation campaign of full size engines it is too early to jump to final conclusions.

SRC	Ms Peeters, the UWV, the Netherlands, Czech, Danish and German Governments and the European Commission submitted written observations to the Court.
REF	M. A. Peeters, az UWV, a holland, a cseh, a dán, és a német kormány, valamint az Európai Bizottság írásbeli észrevételeket terjesztett a Bíróság elé.
SMT	A ms peeters UWV, Hollandia, a cseh, a dán és a német kormány
0.554	és az Európai Bizottság írásbeli észrevételeket a Bíróság elé.
NMT	Peeters úr, az UWV, Hollandia, cseh, dán és német kormány és
0.230	az európai Bizottság írásos észrevételt nyújtott be a bírósághoz.
SRC	This does not mean that by regulating access to public infrastructure, a resource has been transferred (or indeed forgone).
REF	Ez nem jelenti azt, hogy a nyilvános infrastruktúrához való hozzáférés szabályozásával sor kerül forrásátruházásra (vagy valójában -kiesésre).
SMT	Ez nem jelenti azt, hogy szabályozása által való hozzáférés,
0.329	infrastruktúra állami forrásnak át (vagy akár negatív).
NMT	E nem jelenti az, hogy az állami infrastruktúrához való hozzáférés
0.329	szabályozása révén a forrásokat átruházták (vagy).
SRC	On 24 October 2013, the Council adopted Decision 2013/527/CFSP [2] amending and extending the mandate of the EUSR for the Horn of Africa until 31 October 2014.
REF	A Tanács 2013. október 24-én elfogadta az Európai Unió Afrika szarváért felelős EUKK megbízatásának módosításáról és 2014. október 31-ig történő meghosszabbításáról szóló 2013/527/KKBP határozatot [2].
SMT	A Tanács 2013. október 24-én elfogadta a 2013/527/KKBP határozat [2],
0.421	amely az EUKK megbízatását az Afrika szarva térségre vonatkozóan 2014. október 31-ig.
NMT	A tanács 2013. október 24-én elfogadta az Afrika szarváért felelős
0.917	EUKK megbízatásának módosításáról és 2014. október 31-ig történő meghosszabbításáról szóló 2013/527/KKBP határozatot [2].

Fig. 4. Sample translations by the different systems.

7 Conclusions and future work

We have presented our first promising results of machine translating English to Hungarian with neural models. Due to the novelty of this line of work and technology in our working environment there are limitations in the current setups which will be soon overcome and more extensive tests on much larger datasets can be carried out. This will hopefully lead to a machine translation system that can significantly help human translators at the European Commission, even in languages where the current MT@EC system cannot offer sufficient support and is therefore rarely used by the translators.

Neural translation models are robust, flexible and seem to respond more favourably than phrase-based systems to the changes in parameters of the experiments in many respects; they can utilise linguistic information without introducing much complexity in the model, and seem to capture the properties of the training data better. We hope that they will lead to a breakthrough in the translation of difficult language pairs and soon be mature enough to be used even in our production environments. They are easier to train, customise and

manage than a Moses based system and can benefit from advances on neural models from other fields as well. In the multilingual environment of the EC the possibility of multilingual translation with a single model [32] is a promise which needs serious consideration.

In the near future we plan to further improve the precision of language dependent pre-processing tools, decrease the vocabulary by targeting named entities with special pre-processing, further narrow the distance between source and target by for example adding morphological split for the source side, and set up a large scale manual evaluation campaign of full size NMT engines.

Acknowledgements

This work is carried out in the framework of the CEF eTranslation project. The authors would like to thank the Research Group for Mathematical Linguistics and the Research Group for Language Technology at the Research Institute for Linguistics for giving access to their hardware infrastructure for some of the experiments, Attila Novák for developing and providing the core components of the Hungarian morphological analyser and generator, and the reviewers for their comments and suggestions to improve the paper.

References

1. Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Jimeno Yepes, A., Koehn, P., Logacheva, V., Monz, C., Negri, M., Neveol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., Verspoor, K., Zampieri, M.: Findings of the 2016 conference on machine translation. In: Proceedings of the First Conference on Machine Translation, Berlin, Germany, Association for Computational Linguistics (2016) 131–198
2. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal (2015) 1412–1421
3. Bentivogli, L., Bisazza, A., Cettolo, M., Federico, M.: Neural versus phrase-based machine translation quality: a case study. In: Proceedings of EMNLP 2016, Austin, USA (2016)
4. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions. ACL '07, Stroudsburg, PA, USA, Association for Computational Linguistics (2007) 177–180
5. MT@EC: Secure machine translation for the European Union, European Commission, Directorate-General for Translation (2014)
6. Novák, A., Tihanyi, L., Prószéky, G.: The metamorpho translation system. In: Proceedings of the Third Workshop on Statistical Machine Translation. StatMT '08, Stroudsburg, PA, USA, Association for Computational Linguistics (2008) 111–114

7. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems. (2014) 3104–3112
8. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: ICLR. (2015)
9. Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., Makhoul, J.: Fast and robust neural network joint models for statistical machine translation. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Baltimore, Maryland, Association for Computational Linguistics (2014) 1370–1380
10. Cho, K., Van Merriënboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, Association for Computational Linguistics (2014) 1724–1734
11. Jean, S., Firat, O., Cho, K., Memisevic, R., Bengio, Y.: Montreal neural machine translation systems for WMT'15. In: Proceedings of the Tenth Workshop on Statistical Machine Translation, Lisbon, Portugal, Association for Computational Linguistics (2015) 134–140
12. Lu, Z., Li, H., Liu, Q.: Memory-enhanced decoder for neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas (2016) 278–286
13. Zhang, B., Xiong, D., su, j., Duan, H., Zhang, M.: Variational neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, Association for Computational Linguistics (2016) 521–530
14. Almahairi, A., Cho, K., Habash, N., Courville, A.C.: First result on Arabic neural machine translation. CoRR **abs/1606.02680** (2016)
15. Sennrich, R., Haddow, B., Birch, A.: Edinburgh neural machine translation systems for wmt 16. In: Proceedings of the First Conference on Machine Translation, Berlin, Germany, Association for Computational Linguistics (2016) 371–376
16. Crego, J., Kim, J., Klein, G., Rebollo, A., Yang, K., Senellart, J., Akhanov, E., Brunelle, P., Coquard, A., Deng, Y., Enoue, S., Geiss, C., Johanson, J., Khalsa, A., Khiari, R., Ko, B., Kobus, C., Lorieux, J., Martins, L., Nguyen, D.C., Priori, A., Riccardi, T., Segal, N., Servan, C., Tiquet, C., Wang, B., Yang, J., Zhang, D., Zhou, J., Zoldan, P.: SYSTRAN's pure neural machine translation systems. CoRR **abs/1610.05540** (2016)
17. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's neural machine translation system: Bridging the gap between human and machine translation. CoRR **abs/1609.08144** (2016)
18. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, Association for Computational Linguistics (2016) 1715–1725
19. Chung, J., Cho, K., Bengio, Y.: A character-level decoder without explicit segmentation for neural machine translation. In: Proceedings of the 54th Annual Meeting

of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, Association for Computational Linguistics (2016) 1693–1703
20. Tu, Z., Lu, Z., Liu, Y., Liu, X., Li, H.: Modeling coverage for neural machine translation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, Association for Computational Linguistics (2016) 76–85
21. Shen, S., Cheng, Y., He, Z., He, W., Wu, H., Sun, M., Liu, Y.: Minimum risk training for neural machine translation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, Association for Computational Linguistics (2016) 1683–1692
22. Luong, T., Sutskever, I., Le, Q., Vinyals, O., Zaremba, W.: Addressing the rare word problem in neural machine translation. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, Association for Computational Linguistics (2015) 11–19
23. Novák, A., Siklósi, B., Oravecz, Cs.: A new integrated open-source morphological analyzer for Hungarian. In Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., eds.: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), Paris, France, European Language Resources Association (ELRA) (2016)
24. Dyer, C., Chahuneau, V., Smith, N.A.: A simple, fast, and effective reparameterization of IBM Model 2. In: Proceedings of NAACL-HLT, Atlanta, Georgia (2013) 644–648
25. Chen, W., Matusov, E., Khadivi, S., Peter, J.T.: Guided alignment training for topic-aware neural machine translation. In Green, S., Schwartz, L., eds.: Proceedings of The Twelfth Conference of The Association for Machine Translation in the Americas. Volume 1., Austin, Texas (2016) 121–134
26. Sennrich, R., Haddow, B.: Linguistic input features improve neural machine translation. In: Proceedings of the First Conference on Machine Translation, Berlin, Germany, Association for Computational Linguistics (2016) 83–91
27. Lavergne, T., Cappé, O., Yvon, F.: Practical very large scale CRFs. In: Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL), Association for Computational Linguistics (2010) 504–513
28. Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv e-prints **abs/1412.3555** (2014) Presented at the Deep Learning workshop at NIPS2014.
29. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9** (1997) 1735–1780
30. Steinberger, R., Eisele, A., Klocek, S., Pilos, S., Schlüter, P.: DGT-TM: A freely available translation memory in 22 languages. In: Proceedings of the 8th International Conference on Language Resources and Evaluation, Istanbul (2012) 454–459
31. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR. (2015)
32. Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., Dean, J.: Google's multilingual neural machine translation system: Enabling zero-shot translation (2016) arXiv:1611.04558.

Word Embedding-based Task Adaptation from English to Hungarian

Zsolt Szántó, Carlos Ricardo Collazos García, Richárd Farkas

University of Szeged, Institute of Informatics
Árpád tér 2., Szeged, Hungary

Black Swan Data Inc.
Tisza L. krt 47., Szeged, Hungary

{zsolt.szanto, richard.farkas}@blackswan.com

Abstract: In commercial Natural Language Processing (NLP) solutions, we frequently face the problem, that a particular NLP application has to work on several languages. Usually the solution is first developed on a single language – the *source* language – then it is adapted to the other languages – the target languages . In this paper, we introduce experimental results on English to Hungarian adaptation of document classification tasks. In our setting, only an English training dataset is available and our aim is to get a classifier which works on Hungarian documents. We experimented comparatively with two different approaches for word embedding-based language adaptation methods and evaluated them along with monolingual methods in a sentiment classification and a topic classification dataset.

1 Introduction

In commercial Natural Language Processing (NLP) solutions, we frequently face the problem, that a particular NLP application has to work on several languages. Usually the solution is first developed on a single language – the *source* language hereafter – then it is adapted to the other languages – the target languages. There are many opportunities for these adaptations:

1. The necessary resources (like training data labeling and dictionaries) can be constructed manually from scratch to the target language following the principles and best practices recognized during the experiments on the source language. Then we can train models exploiting the brand new resources.
2. The necessary resources can be translated from the source language to the target language then we can train models on these translated resources. Translation can be done manually or by machine translation systems. In both cases it might introduce errors like for example the translation of dictionary items without knowing their purpose/context might be problematic for humans as well.

3. Statistical approaches can be applied for language adaptation itself. Adaptation in this case is not text translation but for example it can be carried out in word embedding spaces.

In this paper, we introduce experimental results with the 3rd approach on English to Hungarian adaptation of document classification tasks. In this setting, only an English training dataset is available and our aim is to get a classifier which solves the same task on Hungarian documents. We comparatively experimented with two different approaches for word embedding-based language adaptation methods and evaluated them, along with monolingual methods, in a sentiment classification and a topic classification dataset.

To the best of our knowledge, this is the first work on automatic language adaptation of any NLP tasks to Hungarian.

2 Word Embedding-based Language Adaptation Techniques

There have been two main approaches published for word embedding-based language adaptation. The earlier approach utilizes a bilingual dictionary to train a mapping between the monolingual word embeddings of the source and target languages [6]. The recent approaches exploit parallel corpora and construct a bilingual word embedding from it [11]. Here, we briefly introduce the principles of word embedding and these two approaches for language adaptation.

2.1 Word Embedding

A word embedding is a distributional representation of words in a few hundred dimensional continuous vector space [7], [10]. Two vectors are close to each other in the embedding space if the words they belong to are similar to each other. More precisely, if two words appear in similar contexts their vector representations are pushed to be close to each other during the construction of the word embedding.

The distributed representations for words have become extremely successful. Their main advantage is that they can help to model unseen or rare words. Usually we train a word embedding on huge unlabeled corpora. On the other hand, the training corpus in a supervised machine learning setting is relatively small. In prediction time, if we find a word which was not present in the training data we can look for similar words in the word embedding thus generalizing the patterns learnt from the training data.

Mikolov et al. [8] also showed that the distributed representations of words capture surprisingly many linguistic regularities, and that there are many types of similarities among words, which can be expressed as linear translations.

There are two popular models for learning word embedding efficiently on large amounts of texts, namely Skip-gram and CBOW [5]. Here, we briefly introduce Skip-gram as it is extended into a bilingual model we use in this paper. In the Skip-gram model [5], the training objective is to learn word vector representations that are good at predicting their context in the same sentence. More formally, given a sequence of

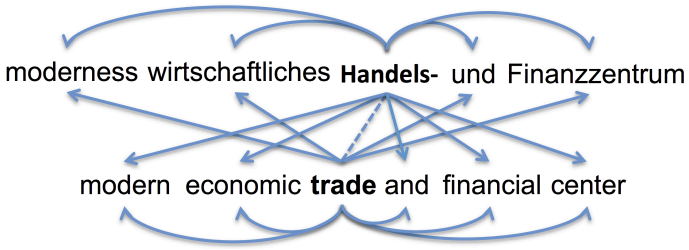


Figure 1. A German and English word aligned phrase to depict the BiSkip model. It exploits monolingual context like skip-gram and also cross-linguality based on the given word alignment [4].

training words, the objective of the Skip-gram model is to maximize the average conditional
probability of the words in a given window conditioned on the middle word. In the Skip-gram model, every word is associated with two learnable parameter vectors, the word vector and the context vector. After training, the context vectors are dropped and word vectors are used as word embedding.

Due to its low computational complexity, the Skip-gram model can be trained on a large corpus in a short time, i.e. billions of words in hours.

2.2 Bilingual Dictionary-based Adaptation

The first attempts at word embedding-based language adaptation were based on two pretrained monolingual word embeddings on both the source and the target languages. Then they learn a mapping, called translation matrix, between the two vector representations exploiting the entries of a bilingual dictionary as training examples [6].

The hypothesis behind this approach is that the vector representations of similar words in different languages are related by a linear transformation. Hence they optimize for a *translation matrix,* which is able to linearly transform any words from the target language's word vector space to the source language's vector space with minimum distance.

For a document classification task, we can train a machine learning model on the available English training corpus utilizing the word vectors from the document. In prediction time, we look for each word vector of the Hungarian document's words, map each of them through the translation matrix into the English word embedding space. Then the machine learnt model makes its prediction on the translated vectors.

2.3 Parallel Corpus-based Adaptation

More recent approaches aim to learn a single joint bilingual word embedding for the source and target languages from a parallel corpus [11]. Their assumption is that by allowing the joint model to utilize both the co-occurrence context information within a

language and the meaning-equivalent signals across languages, they can obtain better word vectors both monolingually and bilingually. We use the so called *BiSkip* bilingual embedding model in this paper because in Upadhyay et al. [11], BiSkip proved to be the most robust and accurate in comparison with other state-of-the art bilingual embedding models.

Luong et al. [4] proposed the Bilingual Skip-Gram (BiSkip) algorithm, an extension of the monolingual skip-gram model, which learns bilingual embeddings by using a parallel corpus along with word alignments (see Figure 1). The learning objective is an extension of the skip-gram model, where the context of a word is expanded to include bilingual links obtained from word alignments, so that the model be trained to predict words cross-lingually.

Figure 2 shows colors in English – Hungarian bilingual vector space. We used PCA to reeducate the dimensions of the vectors.

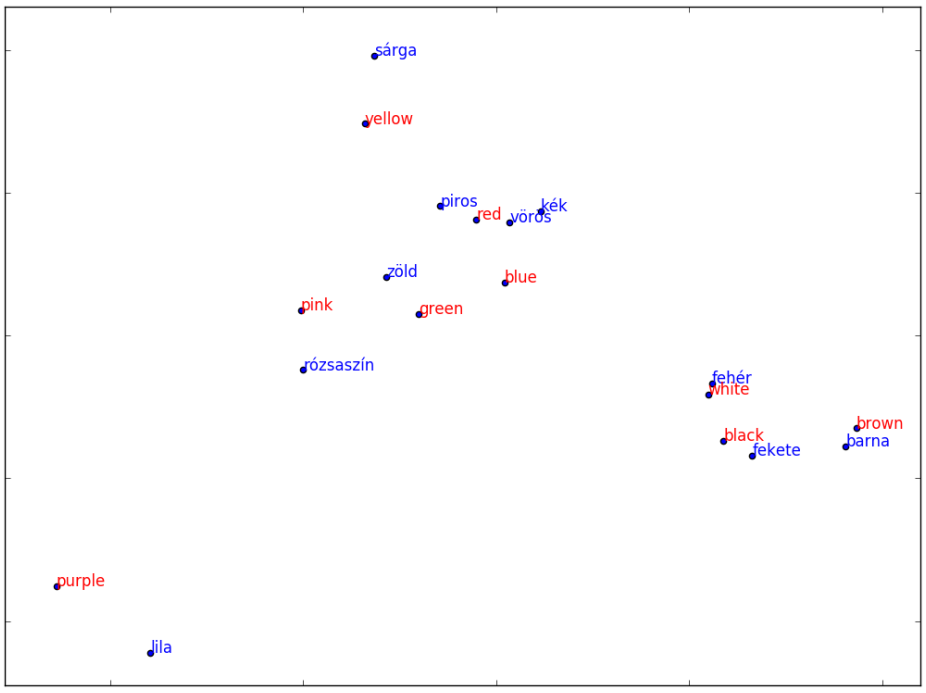


Figure 2. Hungarian and English colors in BiSkip trained vectors.

3 Evaluation Datasets

We evaluated the adaptation approaches in two types of classification tasks, sentiment and topic classification. In each case, we worked on user generated short texts from social media. Both datasets are binary classification problems (i.e. there are two class

labels) and the distribution of the labels is uniform. The sentiment corpora have *positive* and *negative* labels, while the topic classification task contains *game* and *sport* labels.

Table 1 summarizes the sizes of the train and the evaluation datasets for English and Hungarian for the two evaluation scenarios.

Table 1. Sizes of datasets used for evaluating language adaptations.

	Sentiment		topic	
	HU	EN	HU	EN
train #doc	5 000	5 000	10 000	10 000
eval #doc	1 000	1 000	2 000	2 000
train #token	60 945	10 5648	173 655	153 662
eval #token	12 122	20 307	33 274	29 338

3.1 Sentiment Classification

We downloaded product reviews from the English `newegg.com` and the Hungarian `arukereso.hu` sites. The reviews are coming from the IT domain in both languages. Moreover, both sites contain *pro* and *con* fields where a user summarizes his opinion. We used only these summaries and took *pro* as *positive* and *con* as *negative* documents. We removed the too short (less than 4 tokens) documents as they usually hold only placeholder content, like 'none'.

3.2 Topic Classification

In topic classification, our objective was to develop a classifier, which is able to identify the central topic of any user-generated text (like Facebook posts or tweets). For a feasibility study, we downloaded public `Facebook` posts from Hungarian and English sites in the *computer/console game* and *sports* topics through the Facebook Graph API[1]. These Facebook sources are listed in Table 2.

Table 2. The sources of topic classification datasets (Facebook pages)

HU	game	PCGuruMagazin, gamestarhu, 576Kbyte, gamedayiroda
HU	sports	nsonline, focihiradohu
EN	game	kontaku, pcgamemagazin
EN	sports	SkySports, ESPN

[1] https://developers.facebook.com/docs/graph-api/

4 Experimental Setting

In this section, we describe the bytes and bits of our experimental setups for the two adaptation approaches.

4.1 Translation matrix

For the translation matrix approach, we employed the Polygot pretrained embeddings [1]. Polyglot embeddings are publicly available for more than 100 languages trained on Wikipedia dumps of the languages (89 million Hungarian and 1704 million English tokens) using the Skip-Gramm model. Each polyglot model contains the most frequent 100000 words from the selected dump. We used their 64 dimensional vectors for Hungarian and English. The mapping between the embeddings of the two languages was learnt on the Universal dictionary database[2], i.e. we were optimizing a mapping which can achieve the minimum of the sum squared error on mapping Hungarian word vectors of the dictionary to English word vectors. We trained a linear regressor for each of the dimensions and also experimented with Canonical-correlation Aanalysis but they could achieve similar results.

Having the word vector mapping, we train a classifier on the English training dataset then in prediction time, we map the word vectors of the Hungarian document in question into the English word embedding space and carry out the classification based on the mapped vectors. In our experiments, we calculated the average of the word vectors of a document and use these averages as features of a logistic regression classifier (using the `python sklearn` implementation with its default metaparameters [9]). We also tried neural network based approaches, Convolutional Neural Network and Recurrent Neural Networks for exploiting the word vector representations but could not get higher scores.

4.2 BiSkip adaptation

The bilingual word vectors were constructed on 10 million English-Hungarian sentence pairs of the `OpenSubtitles` parallel corpus [3]. We chose this parallel corpus for our experiments because movie subtitles are closer to social media texts than other available parallel corpora as they use more slang and have a conversational nature. First, we calculated word alignment on the parallel corpus with `fast_align`[3] [2] then we trained the BiSkip bilingual word embedding with the `bivec`[4] tool [4], (we used the default parameters, dimension: 200, window size: 5, iterations: 50). The bilingual model contains 298728 Hungarian and 120615 English words.

In this scenario, we train again a logistic regression classifier on the English training dataset using the average of the word vectors as document features. In prediction time,

[2] http://www.dicts.info/uddl.php
[3] https://github.com/clab/fast_align
[4] https://github.com/lmthang/bivec

we take the average of the Hungarian word vectors and apply the classifier on the top of them as they are consistent with the English embedding.

Table 3 shows the ratio of words from the classification datasets which are not in the vocabularies of the word embeddings.

Table 3. Ratio of the out-of-vocabulary words in classifications datasets.

	Sentiment		topic	
	HU	EN	HU	EN
polyglot	10.26%	6.32%	17.70%	8.07%
subtitles	8.00%	5.12%	15.53%	6.76%

5 Results

The baseline in each evaluation setting is 50% accuracy as we always have uniform label distribution and binary classification tasks. We compare the results of word embedding-based adaptation against monolingual results, i.e. against the accuracies that might be achieved if a Hungarian training corpus with the same size would be available. These results can be considered as upper bounds for the language adaptation scenario.

Table 4. Accuracies achieved by various models on two evaluation settings.

	sentiment	topic
bag-of-word	92.3	87.5
translation matrix	52.1	53.0
BiSkip, monoling, 10M sent	88.3	81.8
BiSkip, train on EN, 10M sent	80.1	66.2
BiSkip, monoling, 1M sent	87.1	76.6
BiSkip, train on EN, 1M sent	78.4	54.9

Table 4 consists of the accuracy scores achieved on the Hungarian evaluation datasets. The ‘bag-of-word’ and the ‘monoling’ models are trained on the Hungarian training dataset, hence they are upper bounds for the ‘translation matrix’ and ‘train on EN’ models which have access only to English training data. The bag-of-word model is a logistic regression classifier with uni- and bigram features. The word embedding-based approaches are introduced in the previous section.

We tried the following parameters: word vector sizes 50, 100, 200; number of iterations: 5, 20, 50. The table contains the best results among these parameter settings for each evaluation scenario.

6 Discussion

The monolingual results are an upper bound for the language adaptation experiments but there is a considerable gap between the two monolingual settings, i.e. between monolingual BiSkip and the bag-of-word results achieved. Both approaches train a logistic regression classifier on the Hungarian training dataset. The key difference between them is at the feature representation, which consists of uni- and bigram tokens versus average of word vectors. The reason for this gap may be the size of the training datasets as it might happen that few thousand training examples are sufficient to learn the contribution of particular uni- and bigrams and the average of word vectors becomes to be too general.

Table 3 shows that the translation matrix approach failed in these experimental setups. Most likely, the Polyglot word embedding – trained on the Wikipedia – is not suitable for the distributed representation of social media text. Another explanation might be that the one-to-one translation of words from Hungarian to English is not linear. For instance, it should map each form of a Hungarian noun to the same vector in the English embedding.

BiSkip could achieve much better results. Its bilingual results are fair to compare with its monolingual results as it avoids the bag-of-words versus vector representation effect. The difference between the monolingual and bilingual results are 8 and 15 percentage points on the sentiment and topic classification tasks, respectively. This is the price we have to pay if we do not label a monolingual training dataset but employ a state-of-the-art automatic language adaption technique. A possible reason for the difference between the gaps at the two tasks is that in topic classification named entities are more important than in the sentiment task, and the translation of the named entities are very easy, but if the parallel corpus (which created from subtitles) does not contain a named entity you cannot generate a word vector to it.

Finally, Table 3 also reveals the effect of data amount which the word embedding was calculated on. In each setting, a considerable improvement can be observed if BiSkip can be trained on 10 million sentence pairs instead of 1 million sentence pairs.

7 Conclusions

We introduced experiments with state-of-the-art language adaptation techniques from English to Hungarian. We assume a document classification task where only an English labeled training dataset is available but we aim to solve the same classification task in Hungarian documents. Our experiments on a sentiment and on a topic classification task showed that the translation matrix-based method failed while the BiSkip method could considerably outperform it. Our experiments support that the corpus which the word embedding is trained on and the document classification corpus have to be as lose as possible in domain and that the size of the parallel corpus exploited is important. Our final conclusion is that state-of-the-art language adaptation methods can achieve roughly 10 percentage point worse results compared to the situation where a labeled training corpus would be available.

Acknowledgements

We are grateful for the work of Viktor Hangya on downloading and cleaning the sentiment classification datasets.

The research of Richárd Farkas is supported by the János Bolyai Research Scholarship of the Hungarian Academy of Science.

References

1. Rami Al-Rfou, Bryan Perozzi, Steven Skiena: Polyglot: Distributed Word Representations for Multilingual NLP. In Proc. CoNLL, pp 198-192 (2013)
2. Chris Dyer, Victor Chahuneau, and Noah A. Smith: A Simple, Fast, and Effective Reparameterization of IBM Model 2. In Proc. of NAACL (2013)
3. Pierre Lison, Jörg Tiedemann: OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In Proc. of LREC (2016)
4. Minh-Thang Luong, Hieu Pham, Christopher D. Manning: Bilingual Word Representations with Monolingual Quality in Mind. In Proc. of NAACL-HLT. pp 151-159. (2015)
5. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781. (2013)
6. Tomas Mikolov, Quoc V. Le, and Ilya Sutskever: Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168* (2013)
7. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean: Distributed representations of words and phrases and their compositionality. In Proc. NIPS 26, pp 3111–3119. (2013)
8. Tomas Mikolov, Scott Wen-tau Yih, and Geoffrey Zweig: Linguistic regularities in continuous space word representations. In Proc. of NAACL HLT (2013)
9. Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E.: Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 12, pp. 2825-2830 (2011)
10. David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams: Learning representations by back-propagating errors. *Nature*, 323 (6088):533–536. (1986)
11. Shyam Upadhyay, Manaal Faruqui, Chris Dyer, Dan Roth: Cross-lingual Models of Word Embeddings: An Empirical Comparison. In Proc. of ACL. pp 1661–1670 (2016)

VI. Poszterek

A 2016-os tanártüntetések szövegeinek feldolgozása és adatvizualizációja interaktív dashboard segítségével

Balogh Kitti[1], Fülöp Nóra[1], Szabó Martina Katalin[1,2]

[1]Precognox Informatikai Kft.
kbalogh@precognox.com; mszabo@precognox.com;
noraa.fulop@gmail.com
[2]Szegedi Tudományegyetem, Orosz Filológiai Tanszék
szabo.martina@lit.u-szeged.hu

Kivonat: A dolgozatban egy automatikus emóció- és szentimentelemzéssel, valamint topik modellezéssel feldolgozott korpusz létrehozásáról, valamint az adatokon végzett elemzésekről és vizualizációs megoldásokról számolunk be. A korpuszt olyan posztokból és kommentekből hoztuk létre, amelyek tematikájukban a 2016. februári és márciusi tanártüntetések eseményeihez kapcsolódnak. A szövegeket automatikusan gyűjtöttük le, melyeken aztán szótáralapú szentiment- és emócióelemzést hajtottunk végre, és topik modellezés módszerével nyertünk ki témákat. Az így feltárt szemantikai adatokat végül egy interaktív dashboard segítségével vizualizáltuk.

1 Bevezetés

Jelen dolgozat egy különböző nyelvtechnológiai tartalomelemző módszerekkel feldolgozott korpusz létrehozásáról, valamint a tartalmi adatok vizualizációs megoldásáról, és az azok alapján levonható következtetésekről számol be.

Tárgyaljuk a kutatás során felhasznált eszközöket és módszereket, valamint a feldolgozott korpusz eredményeit egy lehetséges szociológiai megközelítés segítségével. Ismertetjük a saját fejlesztésű adatgyűjtő eszközöket, amellyel lehetőség nyílik Facebookon elérhető adatok gyűjtésére. Emellett bemutatjuk a létrehozott korpuszt, amelynek szövegei tematikájukban a 2016-os tanártüntetések eseményeihez kapcsolódnak, és amelyet szótáralapú automatikus szentiment- és emócióelemzéssel, valamint topik modellezéssel dolgoztuk fel. Az utóbbi módszerek részletes betekintést engednek a korpusz szövegeinek tartalmi sajátságaiba.

A topik modellezés, a szentiment- és az emócióelemzés eredményeinek megértéséhez olyan szociológiai irodalmat használtunk fel, amely a társadalmi mozgalmakban előforduló érzelmekkel foglalkozik. Mindezek mellett a kutatás eredményeit interaktív dashboardon jelenítettük meg, amely vizuális eszközeivel biztosítja az eredmények áttekinthetőségét. A dashboard a http://labs.precognox.com/fbshiny/ linken érhető el.

2 A munka elméleti háttere

A szentimentelemzés segítségével feltárható sajátságokat ma már számos tartalomelemző megoldás hasznosítja [1]. Ennek oka az értékítéletek, vélemények jelentőségének felismerésében keresendő, aminek következtében a szentimentelemzés fontos kutatási és fejlesztési irány az adat- és szövegbányászat területein. Bár a szentimentelemzés mára az egyik legaktívabban művelt NLP-területté nőtte ki magát [2], a magyar nyelvű szövegek értékeléselemzése még mindig jóval kisebb figyelmet kap a nemzetközi gyakorlathoz képest [3], [4], [5], [6], [7].

Az emócióelemzés, a szentimentelemzéstől eltérően nem az értékítéletek, hanem az érzelmek automatikus feldolgozását célozza. Ez alapján, bár a két tartalomelemző feladat között van összefüggés, azok nem azonosíthatóak egymással [1], [8]. Az érzelmek szövegalapú elemzésével még nemzetközi szinten is csekély számú nyelvtechnológiai dolgozat foglalkozik [9], [10]. A magyar nyelvű szövegek érzelemelemzésére pedig alig fordítanak figyelmet a kutatók és fejlesztők [1], [8]. Rá kell mutatnunk azonban, hogy az érzelmek bizonyos tudományos diszciplínákban, így a viselkedéstudományban vagy a pszichológiában központi szerepet töltenek be. Kutatási eredményeink [1], [8] alapján úgy véljük, hogy a szövegek emotív tartalmának kinyerése olyan értékes információkat hozhat a felszínre, amelyeket más tartalomelemző módszerek nem tárnak fel. Ezzel összefüggésben, a jelen korpusz esetében is, a szentimentelemzés mellett emócióelemzést is végeztünk a szövegeken. Mivel a szövegek az oktatás átalakításáért küzdő mozgalmak Facebookon is aktív résztvevőitől származnak, nyomon követhetővé válik, hogy a mozgalomban – ahogy a mozgalmak esetében általánosan – a politikai cselekvés elengedhetetlen velejárói az érzelmek, amelyek segítenek a politikai mozgósításában, a csoporttagok rekrutálásában és az elköteleződés növelésében [12].

A topik modellezés egy nagy népszerűségnek örvendő szövegbányászati módszer, amely nagy mennyiségű szöveges adat rejtett tematikus struktúrájának feltárását célozza [11]. A topik modellek kutatása egy többfelé ágazó, folyamatosan fejlődő terület. Jelen dolgozattal két kutatási irányhoz szeretnénk csatlakozni. Ezek egyike egy a még kevesek által művelt irány, amely a topik modellek eredményeinek interdiszciplináris (szociológiai, jogi, politikai, stb.) felhasználási lehetőségeivel foglalkozik [11]. Kutatásunk során a kinyert témákat, érzelmeket és értékítéleteket szociológiai szempontból értelmezzük. A düh, öröm, meglepettség, félelem, undor és szomorúság mértékének kimutatása mellett fontos eleme az érzelmek vizsgálatának az, hogy mi váltja ki ezeket az érzelmeket: milyen kontextusokban jönnek létre és milyen események vannak rájuk hatással [12]. Ugyanakkor a terület vizsgálata komplex feladat, a megjelenő érzelmek és a különböző témák dinamikus változása módszertanilag nehezen hozzáférhető [12]. A közösségi média oldalairól gyűjthető adatok emóció- és szentimentelemzéssel, valamint topik modellezéssel történő feldolgozása, és az eredmények idősoros követése lehetséges eszközként kínálkozik.

A másik kutatási irány, amelyhez a jelenlegi kutatásunk kapcsolódik, a topik modellek eredményének vizualizációjára és megfelelő user interface-ek létrehozására irányul [11]. E célból hoztunk létre egy interaktív dashboardot, amelyen a vizualizációk a beállításokon keresztül befolyásolhatóak, így a felhasználót érdeklő információk kiemelhetőek A dashboardon a posztok és a kommentek témáit

idősorosan jelenítettük meg, ami lehetőséget ad az eseményekkel kapcsolatos fontos témák időbeli alakulásának vizsgálatára. A felületen a Facebookról legyűjtött két esemény aktivitás- (posztolás, kommentelés, like- és reakcióadás) és szöveges adatainak vizualizációi is megjelennek. A jelenlegi dolgozatban alapvetően a szöveges adatokra koncentráltunk, az aktivitás adatok feldolgozásáról nem számolunk be.

3 A korpusz bemutatása és alapvető adatai

A korpusz annak a két Facebook-eseménynek a posztjait és kommentjeit tartalmazza, amely oldalakon keresztül a 2016-os tanártüntetésekkel kapcsolatos mozgalom szerveződött.

Eszközeink és a téma metszéspontjánál egy termékeny szociológiai terület bontakozik ki, a társadalmi mozgalmak kutatása és a társadalmi mozgalmakban megjelenő érzelmek vizsgálata. A két kiválasztott Facebook eseményt elemezve – amelyeken jelentős felhasználói aktivitás volt mérhető, és a demonstrációkon is sokan vettek részt – reméljük, hogy képet adhatunk egy mozgalom résztvevőiről, céljaikról és beszédtémáikról.

A korpusz posztjai és kommentjei a 2016. február 2. és március 23. közé eső időszakban keletkeztek, amelyeket automatikusan gyűjtöttük a február 13-i és március 15-i események oldaláról.

A teljes korpusz a szövegműfajok tekintetében 6094 posztból és kommentből áll, amelyekben összesen 15878 mondat, 160589 szó és 1201369 karakter található. Tartalmi szempontból ugyancsak két részre bontható, a bennük tárgyalt események alapján (a februári, valamint a márciusi történésekre). A februári esemény vonatkozásában nagyjából kétszer annyi szöveges megnyilvánulást tettek a felhasználók, mint a márciusi esemény oldalán: a februári eseménynél 4082, a márciusi eseménynél 2012 poszt vagy komment keletkezett a vizsgált időszakban. A két esemény alatti kommentek, illetve posztok közel azonos hosszúságúak az átlagos mondat-, szó- és karakterszám tekintetében. Mindkét eseménynél magasabb volt a kommentek száma a posztokénál. A két eseménynél összesen 4785 kommentet és 1309 posztot írtak a felhasználók. Megfigyelhető továbbá, hogy a februári esemény alatt nagyobb arányban érkeztek kommentek a márciusinál: a februári eseménynél körülbelül 4,5-szer, míg a márciusi eseménynél körülbelül 2,5-szer több kommentet találhatunk, mint posztot.

3.1 A szöveggyűjtés módja

A korpusz szövegeit saját Facebook-scraperünkkel gyűjtöttük, amelynek segítségével publikus Facebook-oldalakról nyerhetünk adatokat. Az eszköz nem csupán szöveges adatokat gyűjt, mint például egy adott oldal posztjai és kommentjei, hanem aktivitási adatokat is. A scrapert Python programozási nyelven implementáltuk, ami a publikus Facebook-adatok gyűjtésében egyszerű parancssoros használatot tesz lehetővé. A scraperrel az 1569911826564534 és az 513643295475511 Facebook azonosítóval

rendelkező események oldalairól gyűjtöttük le az adatokat, amelyek letölthetőek voltak az események eltávolításáig. Az utolsó adatokat 2016. március 23-án gyűjtöttük le.[1]

A megfelelő adatvédelmi szempontok figyelembe vétele mellett bármilyen célra szabadon elérhető az eszköz, valamint annak használati módja az alábbi linken: https://github.com/precognox-admin/FBscraper.

4 A szövegek feldolgozása és a szemantikai információk kinyerése

4.1 Nyelvi előfeldolgozás

A legyűjtést követően, a szöveges adatokat UTF-8 karakterkódolású plain text formátumba konvertáltuk, majd a magyarlánc eszközzel [13] nyelvi feldolgozást végeztünk rajtuk. A szövegeket az eszköz tokenizálta, lemmatizálta, valamint azonosította a lemmák szófaját. Ezután a szövegek lemmáit főnevekre, melléknevekre és ismeretlen szófajú elemekre szűrtük.

A szövegekben az emotikonokat és emojikat is kezeltük, a leggyakrabban használatosakat szóalakokra váltottuk át egy általunk bővített szótár segítségével. Például a ":)" emotikont a "simamosoly" szóalakra alakítottuk át. Ezzel a szófajilag szűrt, lemmatizált és emotikonokat, emojikat kezelő kimenettel fogtunk a szemantikai információk kinyerésébe.

4.2 Topik modellezés

Az előfeldolgozott posztokban és kommentekben feltártuk a szövegekben rejlő témákat. Mivel a dashboardot az R statisztikai programnyelv Shiny nevű webapplikációs frameworkjével készítettük el, a topik modell illesztéséhez egy R csomagot, a `topicmodels`-t [14] használtuk.

A korpusz témáinak kinyeréséhez a topik modellek egyik legegyszerűbb tagját, a látens Dirichlet allokációt [16], [17] használtuk, amely az általunk használt csomagban is implementálva van. A modell poszterior eloszlásának közelítéséhez különböző algoritmusokat alkalmaznak, amelyek közül a `topicmodels` csomagban implementált Gibbs mintavételezést használtuk. A korpuszunkban rejlő látens topikok számának meghatározásához egy R-ben implementált függvényt választottunk, a harmonikus átlag módszerét [18], mely már több korpuszunk topik modellezése során bevált [19]. Miután meghatároztuk az optimális témaszámot, unigram alapú modellt illesztettünk a korpuszon. Ezzel megkaptuk minden egyes dokumentum témaeloszlását és a topikok szóeloszlását. A témákat ez alapján, a témákhoz tartozó legjellemzőbb szavak és a témákhoz tartozó legjellemzőbb dokumentumok alapján neveztük el

[1] A Facebook-adatok scrapelésére vonatkozó szabályokról bővebben l. https://www.facebook.com/apps/site_scraping_tos_terms.php

emberi erőforrás segítségével. A februári esemény posztjaiból 15, kommentjeiből 13 témát nyertünk ki, míg a márciusi esemény posztjaiban 13, kommentjeiben pedig 10 témát tártunk fel.

4.3 Szentiment- és emóciótartalom kinyerése

A második lépésben a korpusz szövegeit a saját készítésű szentiment- és emóciószótárainkkal elemeztük [1], [20].

Magyar nyelvű szentimentszótárunkat részben automatikus, részben manuális módszerrel hoztuk létre, magyar nyelvű szövegek automatikus szótáralapú szentimentelemzése céljából [20]. Első lépésben egy angol nyelvű, pozitív és negatív listából álló szentimentszótárat automatikusan magyar nyelvre fordítottunk, majd a fordítás eredményét kézzel ellenőriztük, javítottuk, valamint két szinonimaszótár segítségével bővítettük. A szótár készítése során nem csupán mellékneveket, hanem főneveket, határozószókat és igéket is felvettük, amennyiben úgy ítéltük, hogy az adott nyelvi elemnek inherens negatív vagy pozitív szentimentértéke van. Az így elkészített szótárunk, amelyet plain text formatumban tárolunk UTF-8 karakterkódolással, összesen 1748 pozitív és 5940 negatív szentimentszót tartalmaz. A szótár kutatási célokra szabadon hozzáférhető (http://opendata.hu/dataset/hungarian-sentiment-lexicon).

Az emóciószótáraink létrehozásában ugyancsak egy angol nyelvű lexikonra (*Affective Text*) támaszkodtunk, amelynek automatikusan magyarra fordított anyagát kézzel ellenőriztük, javítottuk és kiegészítettük [1]. Az emóciókifejezések osztályozásában Ekman és Friesen (1969) [21] érzelemkategorizálási rendszerét követtük, tehát azt a hat alapérzelmet vettük alapul, amelyek arckifejezéseit a kutatások alapján kultúrafüggetlenül azonos módon produkáljuk és azonosítjuk. Az alapérzelmek, amelyekre támaszkodva a szótárunk kifejezéseit kategorizáltuk, a következők: öröm, düh, bánat, félelem, undor és meglepődés. Az elkészített szótár a hat kategóriában összesen 1798 emóciókifejezést tartalmaz.

A posztokban és kommentekben megtalálható szentimenteket és emóciókat a `sentiment` R csomag egy a szótárainkkal módosított verziójával azonosítottuk be, amely csomag lehetőséget nyújt a szentimentek és az emóciók szótári illetve gépitanulás-alapú osztályozására is.[2]

5 Az adatok vizualizációja – a dashboard létrehozása és az eredmények értékelése

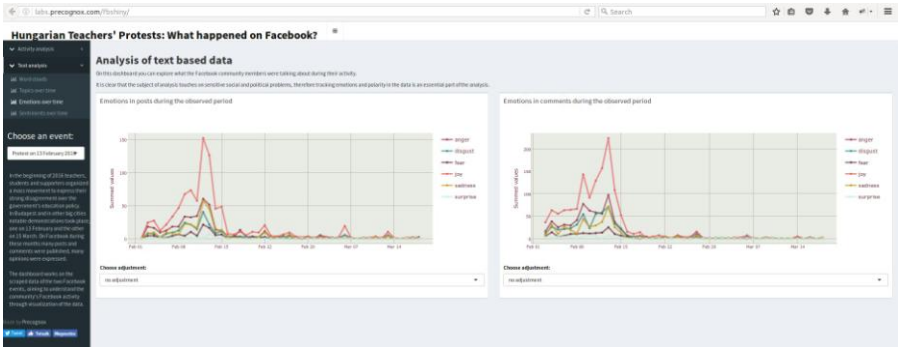
A szövegek többszintű feldolgozását követően a Shiny nevű webapplikációs framework (http://labs.precognox.com/fbshiny/) segítségével készítettünk egy interaktív dashboardot az adatok vizualizálásának és elemzésének céljából.

[2] A csomag a következő címen érhető el: https://github.com/timjurka/sentiment. Az eszköz használatáról többek között l. https://www.r-bloggers.com/sentiment-analysis-with-machine-learning-in-r/.

A http://labs.precognox.com/fbshiny/ linken elérhető dashboard két részből áll. Egyrészt prezentálja és elemzi a két esemény oldaláról nyert aktivitásadatokat, másrészt a szöveges adatokat vizualizálja és elemzi.

A szöveges adatokból kinyert szemantikai információkat négy részre osztva jelenítettük meg a dashboardon. Az első részen a posztok és a kommentek leggyakoribb szavai jelennek meg szófelhő formájában. A februári esemény alatt a többek között a "pedagógus", a "tüntetés", a "gyermek", a "kormány", az "oktatás" és az "ember" szavak jelennek meg hangsúlyosan. Márciusban hasonló képet láthatunk, azonban az 1848. március 15-i események felelevenítése miatt a "március" szó is nagy szerephez jut. A jellemző szavak és a megjelenő topikok mind jól mutatják a mozgalom retorikai készletét, amely döntő fontosságú a célok és a szerepek definiálásában, ezen keresztül pedig a mozgósítás alapjává válik [12].

A társadalmi mozgalmak vizsgálatának fontos szempontja a politikai lehetőségek érzékelése [23], hiszen ez nagy hatással van a tagok elköteleződésére. Az idősorokon keresztül tetten érhetőek azok a témák, amelyek láthatóan kiugrásokat okoztak az érzelmek és szentimentek adataiban. A szöveges adatokban megjelenő szentimentek idősorainál megfigyelhető, hogy mindkét eseménynél a tüntetés időpontjában, illetve pár nappal előtte van a legtöbb negatív és pozitív szentimentet tartalmazó poszt vagy komment, igazolva a politikai cselekvések érzelmileg felfokozott állapotát. A posztok számával arányosított idősorokon más jellegű eseményekre is felfigyelhetünk. Például a februári idősor esetében március 6-án egy nagy negatív csúcs több olyan posztot jelez, amelyben a posztot írók a kormánnyal szembeni ellenérzésüket fejezik ki.



1. ábra. A dashboardon található emóció-idősorok

Az emóciós idősorok alapján a februári eseménynél a posztokban és a kommentekben végig az öröm a legerőteljesebb érzelem, míg a második legjellemzőbb az esemény időpontja körül a szomorúság, majd átveszi a szerepet az undor. A márciusi eseménynél szintén az öröm volt a legerőteljesebb emóció, azonban a második legjellemzőbbnek a düh bizonyult. A szakirodalom szerint az öröm jellemző a társadalmi mozgalmakra, hiszen a tagok így fejezik ki az összefogással kapcsolatos érzelmeiket, amely fontos összetartó erővel bír a kollektíva egészére nézve [12]. Ha azonban az elvárt lehetőségek nem következnek be, könnyen az elkeseredettség vetheti fel fejét, valamint a váratlan csalódásra válaszként a düh [12]. Az eredmények színes képet festenek a tanártüntetések Facebook eseményeinek

érzelmi ökonómiájáról, azonban tekintettel kell lennünk arra is, hogy az eredményeket befolyásolhatják az egyes emóciókhoz tartozó szótárak hosszúságai.

A topik modell illesztésének eredményei átfogó képet adnak a két esemény természetéről. A februári esemény posztjait, kiváltképp közvetlenül a demonstráció előtt és után meghatározó téma a közszolgáltatási szférában dolgozók helyzete. Az oktatás az egészségügy helyzetével összefonódva jelenik meg. Az oktatáspolitikai követelések mellett szó van béremelési követelésekről mind a két ágazatban dolgozók számára, valamint megszólítanak más, a szociális rendszertől függő rétegeket is (nyugdíjasok). Ahogy a posztokban, úgy a kommentekben is láthatóak kormányellenes megnyilvánulások. A februári kommentekre jellemző, hogy a magyar politikai életet tematizálják, és ellenzik a politikusok beleszólását az oktatásügybe. Emellett, talán meghatározóbban az érintettekről szólnak a kommentek: a gyerekekről, a tanárokról, a szülőkről. A februári esemény oldalán jellemzően reálpolitikai követeléseket tematizálnak, egyetlen szimbolikus elem jelenik meg, a szabadság, amely azonban a szabad iskoláztatáshoz kötődik. Mindezzel ellentétben a márciusi esemény oldalán kevés reális követelést olvashatunk, a tüntetést egy történelmi-politikai narratívába helyezik, az 1848-as forradalomhoz hasonlítják, a nép és hatalom konfliktusa váltja fel a szakpolitikai követeléseket. Igaz, a kommentek között megjelennek a gyerekek érdekei (szembeállítva a kormány érdekeivel), de az eredeti konfliktus háttérbe szorul. Mind a két esemény kapcsán megfigyelhető a magyar kultúra szimbolikus használata: sok irodalmi művet és történelmi eseményt idéznek vagy említenek a felhasználók.

6 Összegzés

A dolgozatban egy korpusz létrehozásáról számoltunk be, amelynek anyagát automatikus emóció- és szentimentelemzéssel, valamint topik modellezéssel dolgoztuk fel. Az eredményeket egy interaktív dashboard segítségével vizualizáltuk és elemeztük a megfelelő társadalomtudományi irodalom figyelembe vétele mellett.

A korpuszt olyan posztokból és kommentekből hoztuk létre, amelyek tematikájukban a tanártüntetések eseményeihez kapcsolódnak, és a 2016. február 2. és március 23. közé eső időszakban keletkeztek. Az adatokat egy saját fejlesztésű scraperrel automatikus módszerrel gyűjtöttük, majd a magyarlánc eszközzel dolgoztuk fel. Ezt követően a korpuszon szótáralapú automatikus szentiment- és emócióelemzést hajtottunk végre, és topik modellezés módszerével kinyertünk azok témáit is. A korpusz így feltárt szemantikai adatait végül egy interaktív dashboard segítségével vizualizáltuk.

A létrehozott interaktív dashboard szabadon hozzáférhető az alábbi linken: http://labs.precognox.com/fbshiny/

Köszönetnyilvánítás

A jelen kutatás Az Emberi Erőforrások Minisztériuma Új Nemzeti Kiválóság Programjának támogatásával valósult meg.

Hivatkozások

1. Szabó M.K., Morvay G.: Emócióelemzés magyar nyelvű szövegeken. In Gecső T., Sárdi Cs. (szerk.). Nyelv, kultúra, társadalom. Segédkönyvek a nyelvészet tanulmányozásához 177. Budapest, Tinta (2015) 286-292.
2. Liu, B.: Sentiment Analysis and Opinion Mining. Draft (2012) Elérhető: http://www.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf.
3. Berend, G., Farkas, R.: Opinion Mining in Hungarian based on textual and graphical clues. Proceedings of the 8th conference on Simulation, modelling and optimization. Stevens Point, Wisconsin, USA, World Scientific and Engineering Academy and Society (WSEAS) (2008) 408-412.
4. Miháltz, M.: OpinHu: online szövegek többnyelvű véleményelemzése. In Tanács, A., Vincze, V. (szerk.). VII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2010). Szegedi Tudományegyetem, Szeged (2010) 14-23.
5. Hangya V., Farkas R.,Berend G.: Entitásorientált véleménydetekció webes híranyagokból. In Tanács A., Varga V., Vincze V. (szerk.). XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015). Szeged, Szegedi Tudományegyetem (2015) 227-234.
6. Szabó M. K., Vincze V.: Egy magyar nyelvű szentimentkorpusz létrehozásának tapasztalatai. In Tanács, A., Varga, V., Vincze, V. (szerk.) XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015). Szegedi Tudományegyetem, Szeged (2015) 219-226.
7. Szabó M.K., Vincze V., Simkó K., Varga V., Hangya V.: A Hungarian Sentiment Corpus Manually Annotated at Aspect Level. Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Portoroz, Szlovénia Portoroz: European Language Resources Association (ELRA) (2016) 2873-2878.
8. Szabó M.K., Vincze V., Morvay G.: Magyar nyelvű szövegek emócióelemzésének elméleti nyelvészeti és nyelvtechnológiai problémái. In Reményi A. Á., Sárdi, Cs., Tóth, Zs. szerk. Távlatok a mai magyar alkalmazott nyelvészetben. Budapest: Tinta (2016)
9. Strapparava, C., Mihalcea, R.: Learning to identify emotions in text. SAC 2008. http://web.eecs.umich.edu/~mihalcea/papers/strapparava.acm08.pdf.
10. Mulcrone, K.: Detecting Emotion in Text. Elhangzott: UMM CSci Senior Seminar Conference. Amerikai Egyesült Államok, University of Minnesota: Morris. 2012. április 28. https://wiki.umn.edu/pub/UmmCSciSeniorSeminar/Spring2012Talks/KaitlynM ulcrone.pdf
11. Blei., D.: Probabilistic topic models. Communications of the ACM. 55(4) (2012) 77-84.
12. Goodwin, Jeff; Jasper, James M. Emotions and Social Movements. In: J. E. Stets, J. H. Turner szerk. Handbook of the Sociology of Emotions (2007) 611-631.
13. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A Toolkit for Morphological and Dependency Parsing of Hungarian. Proceedings of RANLP 2013 (2013) 763-771.
14. Grün, B., Hornik, K.: topicmodels: An R Package for Fitting Topic Models. Journal of Statistical Software. 40(13) (2011) 1-30. http://www.jstatsoft.org/v40/i13/.
15. McCallum, A. K. MALLET: A Machine Learning for Language Toolkit (2002) http://mallet.cs.umass.edu.
16. Griffiths, T. L., Steyvers, M.: Finding scientific topics. Proceedings of the National Academy of Sciences (2004) 5228-5235.
17. Blei, D., Ng, A. and Jordan, M. Latent Dirichlet allocation. Journal of Machine Learning Research, 3 (2003) 993-1022.
18. Ponweiser, M.: Latent Dirichlet Allocation in R. Diploma Thesis. Institute for Statistics and Mathematics, Wirtschaftsuniversität Wien (2012)
19. Balogh, K.: A látens Dirichlet allokáció társadalomtudományi alkalmazása. A kuruc.info romaellenes megnyilvánulásainak tematikus elemzése. Szakdolgozat. Survey Statisztika mesterképzés, Eötvös Loránd Tudományegyetem (2015) Elérhető:

http://labs.precognox.com/kurucinfo_adatviz/A_latens_Dirichlet_allokacio_tarsadalomtudomanyi_alkalmazasa_Balogh_Kitti.pdf.
20. Szabó M.K.: Egy magyar nyelvű szentimentlexikon létrehozásának tapasztalatai és dilemmái. In Gecső T., Sárdi Cs. szerk. Nyelv, kultúra, társadalom. Segédkönyvek a nyelvészet tanulmányozásához 177 (2015) 278-285.
21. Ekman, P., Friesen, W.V.: The repertoire of nonverbal behavior: Categories, origins, usage, and coding. Semiotica 1 (1969) 49-98.
22. Cho, M., Schweickart, T., Haase, A.: Public engagement with nonprofit organizations on Facebook. Public Relations Review, 40(3) (2014) 565-567.
23. Goodwin, J., Jasper, J. M.;, Polletta, F.: Return of the Repressed the Fall and Rise of Emotions in Social Movement Theory. Mobilization: An International Journal. 5(1) (2000) 65-83.

Folytonos paraméterű vokóder rejtett Markov-modell alapú beszédszintézisben – magyar nyelvű kísérletek 12 beszélővel

Csapó Tamás Gábor, Németh Géza

Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék,
e-mail: {csapot,nemeth}@tmit.bme.hu

Kivonat A jelen cikkben egy vokódert mutatunk be, amely a beszédet folytonos paraméterekként reprezentálja. A módszer a szakirodalomban ismert parametrikus vokóderekhez képest két fő tulajdonságban különbözik: 1) a zöngés és zöngétlen szakaszokat egységesen (a gerjesztőjel explicit megkülönböztetése nélkül) kezeljük időtartományban egy folytonos alapfrekvencia-mérő algoritmus használatával, 2) a gerjesztőjelet frekvenciatartományban zöngés és zöngétlen komponensek összegeként állítjuk elő, melyeket egy maximális zöngésségi frekvencia érték határol. A vokóder így csak folytonos paramétereket alkalmaz, ami a statisztikai modellezés szempontjából kedvező. Mivel a szintézis rész számításigénye alacsony, ezért a javasolt vokóder hatékonyan alkalmazható korlátozott erőforrású eszközökön is (pl. Android okostelefon) rejtett Markov-modell alapú beszédszintézisben. Az új vokódert beszélő adaptációban is teszteltük, mellyel tetszőleges beszélőre emlékeztető beszédszintetizátor hangot tudunk létrehozni.

Kulcsszavak: gépi tanulás, beszédtechnológia, statisztikai modellezés

1. Bevezetés

A gépi szövegfelolvasás (TTS, Text-To-Speech) egyik legkorszerűbb technológiája a statisztikai parametrikus beszédszintézis [1]. A beszédtechnológiában a statisztikai parametrikus módszerekhez gyakran alkalmazzák a rejtett Markov-modelleket (HMM) [2,3]. Zen és társai szerint három fő területen van kutatásra szükség ahhoz, hogy a statisztikai parametrikus TTS módszerek a természeteshez közeli beszédet eredményezzenek: 1) új típusú vokóderek, 2) az akusztikai modellek pontossága, 3) és a paraméterek túlsimítottsága [1]. Jelen cikkben az első területtel foglalkozunk.

1.1. Vokóderek a statisztikai parametrikus beszédszintézisben

A szakirodalomban számos beszédkódoló módszerről olvashatunk, melyeknek eredeti célja a beszéd paraméterekre bontása (kódolás, analízis lépés) azért, hogy

a távközlési csatornán minél kisebb sávszélesség mellett lehessen átvinni a jelet (beszédet) [4, 244. o.]. Az átvitel után, a vevő oldalon a paramétereket visszaalakítják beszédjellé (dekódolás, szintézis lépés). A parametrikus kódolók, azaz vokóderek családjába tartozik az LPC (Linear Predictive Coding) kódoló, valamint ennek továbbfejlesztett változatai, melyek az elsődleges cél mellett alkalmasak a beszédjel tulajdonságainak változtatására is (pl. F0 módosítás).

Az elmúlt évtizedekben számos vokóder típust kidolgoztak, melyeket a következő kategóriákba sorolhatunk: kevert gerjesztés [5], glottális forrás alapú módszerek [6,7,8], harmonikus-zaj alapú módszerek [9] és maradékjel alapú módszerek [10,11,12] (teljes összehasonlítás: [13, Introduction]). Mindegyik fenti vokódernek az a célja, hogy a HMM-TTS korai változataiban alkalmazott impulzus-zaj elvű vokóder robotosságát, gépiességét, 'zizegését' csökkentsék. Ugyan léteznek olyan vokóderek, melyek közel természetes beszédet tudnak visszaállítani, de ezek tipikusan magas számításigényűek, és ezért nem alkalmazhatóak valós időben (pl. STRAIGHT, [14]).

1.2. A jelen kutatás

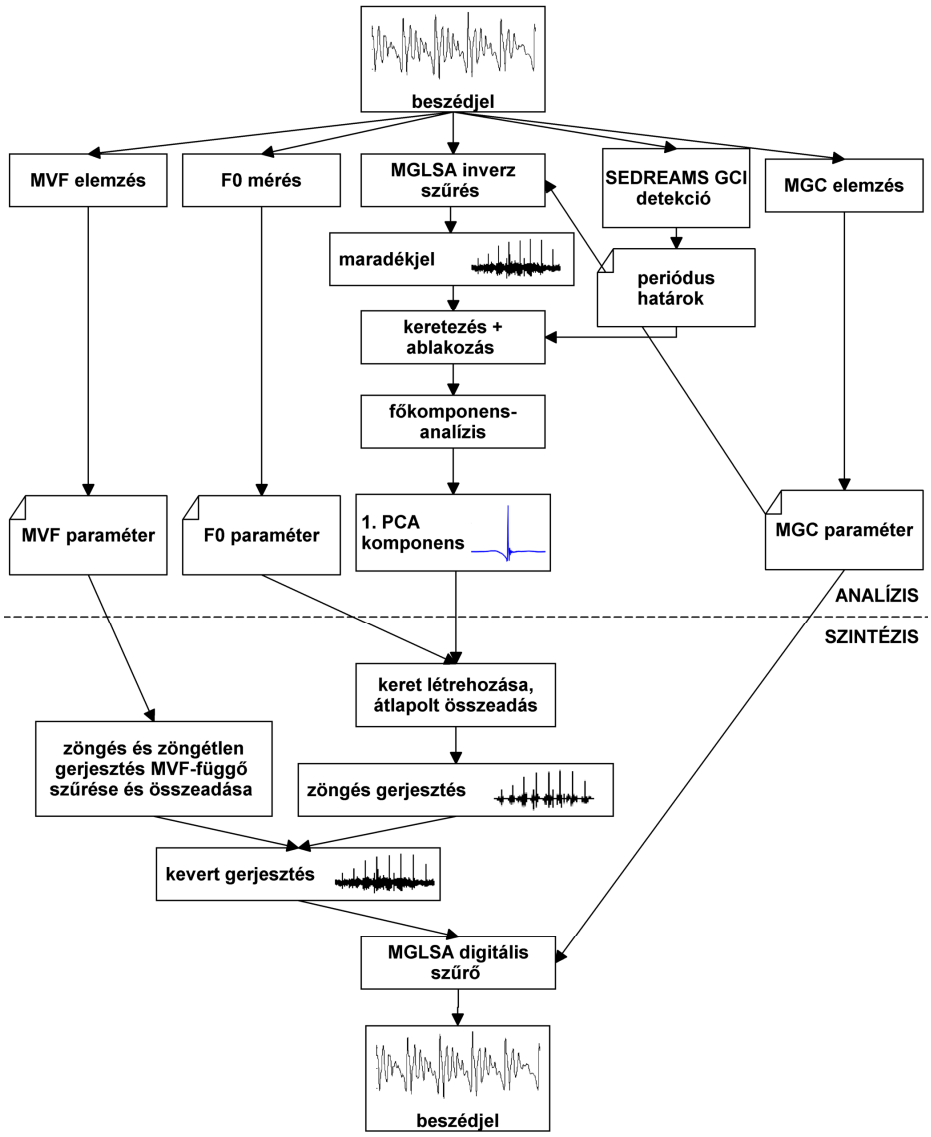
A jelen cikkben egy alacsony komplexitású és számításigényű vokódert mutatunk be. A vokóder korábbi változata maradékjel alapú, és folytonos alapfrekvenciát valamint maximális zöngésségi frekvenciát alkalmaz a zöngés és zöngétlen beszédhangok egységes modellezésére [15]. Később ezt tovább javítottuk a zöngétlen hangok frekvenciakomponenseinek optimális súlyozásával [16]. A mostani cikkben csak a vokóder legutolsó változatát ismertetjük [13] és az erre épülő beszédszintézis alkalmazásokat (magyar nyelvű beszédszintézis Android okostelefonon; TTS adott beszélőre adaptálása néhány percnyi hangminta alapján) is bemutatjuk.

2. Folytonos paraméterű vokóder

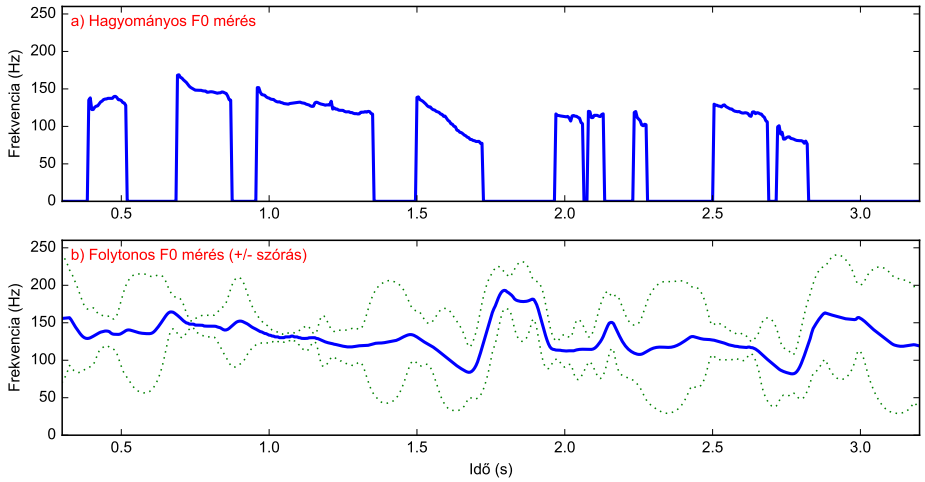
A vokóder analízis és szintézis részekből áll. Az analízis lépés a beszédjel alapján gerjesztési- és spektrális paramétereket állít elő, melyeket a rejtett Markov-modell alapú beszédszintézis modelljeinek betanításához lehet felhasználni. A HMM modell eredményeképpen tetszőleges bemeneti szöveghez generálni tudjuk a gerjesztési- és spektrális paramétereket, majd a vokóder szintézis lépésében a beszéd visszaállítható ezekből.

2.1. Analízis

Az analízis lépéseit az 1. ábra szaggatott vonal feletti része mutatja. Az analízis rész bemenete beszéd hullámforma, amelyet 7,6 kHz-es aluláteresztő szűrés után 16 kHz mintavételezéssel és 16 bites lineáris PCM kvantálással tárolunk. A beszédjelen egy folytonos alapfrekvencia detektorral [17,18] 5 ms eltolással kiszámítjuk az F0 paramétert (F0cont). Ez az F0 detektor a zöngétlen szakaszokon interpolálja az F0-t és Kálmán-szűrést alkalmaz, melynek eredményére a



1. ábra. Analízis (a szaggatott vonal felett) és szintézis (a szaggatott vonal alatt) a folytonos paraméterű vokóderrel.



2. ábra. Az F0 mérés eredménye a) a Snack hagyományos F0 számító algoritmussal [21], b) az SSP folytonos F0 számító algoritmussal [17,18]. A kék folytonos vonal az F0 kontúr, míg a zöld pontozott vonalak a +/- szórást jelölik.

2. ábra mutat példát. Ezután a 'maximális zöngésségi frekvencia' (Maximum Voiced Frequency, MVF, [19]) számítása következik. A következő lépésben spektrális elemzést végzünk 'mel-általánosított kepsztrum' (Mel-Generalized Cepstrum, MGC, [20]) módszerrel . Az elemzéshez 24-ed rendű MGC-t számítunk $\alpha = 0{,}42$ és $\gamma = -1/3$ értékekkel. Végül az MGLSA inverz szűréssel kapott maradékjel zöngeszinkron periódusaiból főkomponens-analízisével kinyerünk egy a későbbi szintézishez használható gerjesztőjelet ('PCA maradékjel', részletek: [15]).

2.2. Az új vokóder rejtett Markov-modell alapú beszédszintézisben

Az analízis résznél leírt paramétereket (F0cont, MVF és MGC) kiszámítjuk a tanító beszédadatbázis mondatainak minden keretére, 5 ms-os eltolással. Az F0cont és MVF paramétereket logaritmizáljuk, majd az MGC-vel együtt a derivált és második derivált értékeket is eltároljuk a paraméterfolyamban. Mivel a paraméterek folytonosak (azaz nincs bennük szakadás, mint a hagyományos F0 kontúr esetén), a modellezés hagyományos HMM-ekkel történik. A tanítás többi része (pl. környezetfüggő címkézés, döntési fák, időtartamok modellezése) a HTS-HUN rendszerrel megegyező módon történik [2,22].

2.3. Szintézis

A szintézis lépéseit az 1. ábra szaggatott vonal alatti része mutatja be. A szintézis bemenetei az analízis eredménye után gépi tanulással modellezett paraméterek (F0cont, MVF és MGC) illetve a 'PCA maradékjel'. A visszaállítás során először a 'PCA maradékjelet' átlapoltan összeadjuk az F0cont-tól függő

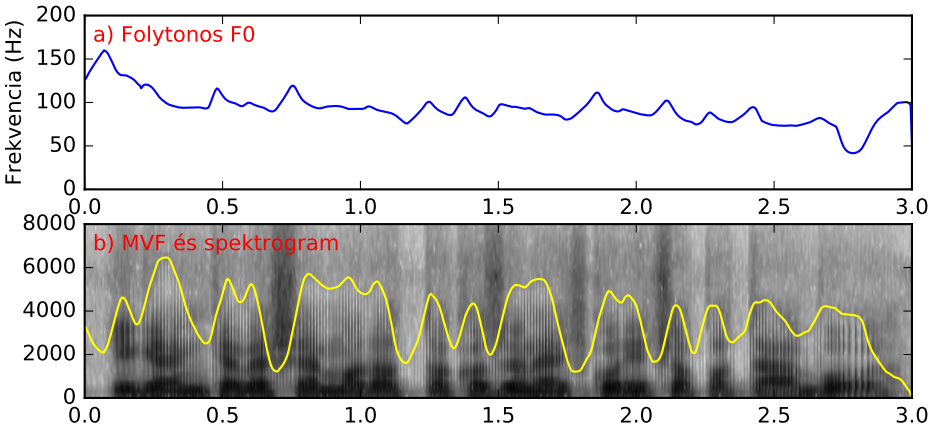
távolságra, ami a gerjesztés zöngés komponensét adja meg. A zöngétlen komponenst fehérzajból hozzuk létre. Mivel nincs külön zöngés/zöngétlen paraméter folyam, az MVF paraméter modellezi a zöngésségi információt, melyet a 3. ábra mutat: a zöngétlen beszédhangok esetén az MVF általában alacsony (200–500 Hz körüli), a zöngés beszédhangoknál magas (tipikusan 4 kHz fölötti), míg a kevert gerjesztésű beszédhangoknál a két véglet közötti (pl. zöngés réshangok esetén 2–3 kHz közötti). A zöngés gerjesztést keretenként az MVF-től függő aluláteresztő szűrővel, míg a zöngétlen gerjesztést felüláteresztő szűrővel módosítjuk, majd összeadjuk a két gerjesztési komponenst. Végül a a szintetizált beszédet az összeadott kevert gerjesztés alapján előállítjuk MGLSA szűréssel az MGC paramétereket felhasználva [23]. Az így szintetizált beszédre a 3. ábra mutat egy példát.

3. Kísérletek és eredmények

3.1. Beszélőfüggő tanítás 12 beszélővel és átlaghang

A kísérletek során magyar nyelvű mintákon végeztük a HMM-ek tanítását és minta szövegek szintézisét. Ehhez a nyelvspecifikus lépéseket a HTS-HUN rendszerből kiindulva alkalmaztuk [22]. A PPBA adatbázis [24,25] hat férfi és hat női beszélőjének hanganyagával végeztünk beszédszintézis kísérleteket. Ehhez a teljes, kb. 2 órányi (beszélőnként közel 2000 mondat) beszédfelvételt és a hozzá tartozó címkézést használtuk fel beszélőfüggő tanítás keretében.

A beszélőfüggő kísérletek után átlaghangot [26] is készítettünk az új vokóderrel és a HTS-HUN rendszerrel. Ehhez a PPBA adatbázis 10 beszélőjét használtuk fel három különböző módon: 1) a tíz beszélőtől származó átlaghang, 2) öt férfi beszélőtől származó átlaghang, 3) öt női beszélőtől származó átlaghang.



3. ábra. Szintetizált beszédminta egy férfi beszélőtől: *'Igen kevesen maradtak az Ön egykori csapatából.'*

1. táblázat. A meghallgatásos teszt eredménye.

Férfi beszélők								Női beszélők							
	1.	2.	3.	4.	5.	6.	7.		1.	2.	3.	4.	5.	6.	7.
FF1	0	0	1	1	0	2	1	NŐ1	0	0	0	0	1	1	3
FF2	1	1	0	3	0	0	0	NŐ2	0	1	1	1	0	2	0
FF3	**4**	1	0	0	0	0	0	NŐ3	0	**3**	0	1	1	0	0
FF4	0	0	1	0	1	1	2	NŐ4	0	1	0	0	3	1	0
FF5	0	2	0	1	2	0	0	NŐ5	**5**	0	0	0	0	0	0
FF6	0	0	0	0	1	2	2	NŐ6	0	0	1	1	0	1	2
FF_átlag	0	1	3	0	1	0	0	NŐ_átlag	0	0	3	2	0	0	0

3.2. Meghallgatásos teszt

A 12 beszélőtől valamint a férfi és női átlaghangból 100–100 mondatot szintetizáltunk, majd egy bekezdést kiválasztottunk egy internetes meghallgatásos teszthez. A tesztelők feladata az volt, hogy ugyanazon mondatokat meghallgatva az összes beszélőtől eldöntsék, hogy melyik férfi és melyik női bemondót preferálják (azaz sorba kellett állítani a beszélőket aszerint, hogy melyik hangkarakter tetszett a legjobban). A preferenciatesztben 5 beszédtechnológiai szakértő vett részt (30–70 év közötti férfiak). Az eredményeket az 1. táblázat mutatja, mely szerint a nők közül NŐ5 és NŐ3 az előnyben részesített, míg a férfiak közül FF3. Az előbbinek az lehet az oka, hogy a preferált női beszélők professzionális bemondók, így az ő hangjuk várhatóan előnyösebb éles TTS rendszerben.

3.3. Beszélő adaptáció

Készítettünk egy Android okostelefonos alkalmazást, amely új beszélőktől hangminták gyűjtésére alkalmas. Öt beszélőtől gyűjtöttünk ilyen módon okostelefonon / tableten felolvasott hangmintákat (50–50 mondatot), majd beszélő adaptációt [22,26] indítottunk az átlaghangokat felhasználva (3.1. fejezet). Az informális meghallgatások szerint az 5 beszélős átlaghangokkal adaptált minták jobban emlékeztetnek az eredeti beszélőre, mint a 10 beszélős átlaghanggal adaptáltak, valószínűleg azért, mert a külön férfi és női beszélőkből álló átlaghangok jobban megőrzik az adott nem jellemzőit.

3.4. Androidos implementáció

Az új vokódert a HTS-HUN rendszer alacsony erőforrású eszközökre optimalizált változatához illesztettük [27]. A HMM-TTS az új vokóderrel közel valós időben (néhány 10 ms-on belül) képes szövegből beszédet szintetizálni átlagos Androidos telefonokon. Precíz meghallgatásos tesztet nem végeztünk az okostelefonokon, de a tapasztalatok szerint az új, folytonos paraméterű vokóderrel kellemesebb beszéd szintetizálható, mint a HTS rendszer egyszerű impulzus-zaj gerjesztésű vokóderével. Korábbi internetes percepciós tesztekben már igazoltuk, hogy az új vokóder természetesebb, mint az alaprendszer [13,15,16].

4. Következtetések

Kutatásunk eredményei számos beszédtechnológiai alkalmazásban felhasználhatóak, amelyek egyrészt hozzájárulhatnak a természetesebb ember-gép kommunikációhoz, másrészt segíthetnek megérteni az emberi beszédképzés működését. A bemutatott beszédszintetizátor rendszer javítja a korlátozott erőforrású eszközökben (pl. Android okostelefon) alkalmazott gépi szövegfelolvasás minőségét. A kevés erőforrás miatt bonyolultabb gerjesztési modellek nehézkesen kezelhetőek, viszont a legújabb vokóder a korlátozott erőforrású eszközökön is képes közel valós idejű beszédszintézisre. A beszédsérülteket segítő kommunikációs eszközökben hasznos lehet, ha a rendszer az eredeti beszélőre emlékeztető hangon szólal meg, amit a beszélő adaptációval oldhatunk meg.

Köszönetnyilvánítás

A kutatást részben támogatta a SCOPES projekt (SP2: SCOPES project on speech prosody, SNSF no IZ73Z0_152495-1) és a VUK (AAL-2014-1-183) projekt keretében az Európai Unió és a Nemzeti Kutatási, Fejlesztési és Innovációs Alap.

Hivatkozások

1. Zen, H., Tokuda, K., Black, A.W.: Statistical parametric speech synthesis. Speech Communication **51**(11) (2009) 1039–1064
2. Tóth, B.P.: Rejtett Markov-modell alapú gépi beszédkeltés. PhD disszertáció, BME TMIT (2013)
3. Tóth, B.P., Németh, G.: Rejtett Markov-modell alapú szövegfelolvasó adaptációja félig spontán magyar beszéddel. In: Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2009), Szeged, Magyarország (2009) 246–256
4. Németh, G., Olaszy, G., eds.: A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Akadémiai Kiadó, Budapest (2010)
5. Yoshimura, T., Tokuda, K.: Mixed excitation for HMM-based speech synthesis. In: Proc. Eurospeech, Aalborg, Denmark (2001) 2263–2266
6. Cabral, J.P., Renals, S., Yamagishi, J., Richmond, K.: HMM-based speech synthesiser using the LF-model of the glottal source. In: Proc. ICASSP, Prague, Czech Republic (2011) 4704–4707
7. Degottex, G., Lanchantin, P., Roebel, A., Rodet, X.: Mixed source model and its adapted vocal tract filter estimate for voice transformation and synthesis. Speech Communication **55**(2) (2013) 278–294
8. Raitio, T., Suni, A., Vainio, M., Alku, P.: Comparing glottal-flow-excited statistical parametric speech synthesis methods. In: Proc. ICASSP, Vancouver, Canada (2013) 7830–7834
9. Erro, D., Sainz, I., Navas, E., Hernáez, I.: Improved HNM-based Vocoder for Statistical Synthesizers. In: Proc. Interspeech, Florence, Italy (2011) 1809–1812
10. Drugman, T., Dutoit, T.: The Deterministic Plus Stochastic Model of the Residual Signal and its Applications. IEEE Transactions on Audio, Speech and Language Processing **20**(3) (2012) 968–981

11. Drugman, T., Raitio, T.: Excitation Modeling for HMM-based Speech Synthesis: Breaking Down the Impact of Periodic and Aperiodic Components. In: Proc. ICASSP, Florence, Italy (2014) 260–264
12. Wen, Z., Tao, J.: Amplitude spectrum based Excitation model for HMM-based Speech Synthesis. In: Proc. Interspeech, Portland, Oregon, USA (2012) 1428–1431
13. Csapó, T.G., Németh, G., Cernak, M., Garner, P.N.: Parametric Vocoder with Continuous F0 Modeling and Residual-based Excitation for Speech Synthesis. submitted to Speech Communication (2017)
14. Kawahara, H., Masuda-Katsuse, I., de Cheveigné, A.: Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds. Speech Communication **27**(3) (1999) 187–207
15. Csapó, T.G., Németh, G., Cernak, M.: Residual-Based Excitation with Continuous F0 Modeling in HMM-Based Speech Synthesis. In Dediu, A.H., Martín-Vide, C., Vicsi, K., eds.: Lecture Notes in Artificial Intelligence. Volume 9449. Springer International Publishing, Budapest, Hungary (2015) 27–38
16. Csapó, T.G., Németh, G., Cernak, M., Garner, P.N.: Modeling Unvoiced Sounds In Statistical Parametric Speech Synthesis with a Continuous Vocoder. In: Proc. EUSIPCO, Budapest, Hungary (2016) 1338–1342
17. : Speech Signal Processing - a small collection of routines in Python to do signal processing [Computer program] (2015) https://github.com/idiap/ssp.
18. Garner, P.N., Cernak, M., Motlicek, P.: A simple continuous pitch estimation algorithm. IEEE Signal Processing Letters **20**(1) (2013) 102–105
19. Drugman, T., Stylianou, Y.: Maximum Voiced Frequency Estimation : Exploiting Amplitude and Phase Spectra. IEEE Signal Processing Letters **21**(10) (2014) 1230–1234
20. Tokuda, K., Kobayashi, T., Masuko, T., Imai, S.: Mel-generalized cepstral analysis - a unified approach to speech spectral estimation. In: Proc. ICSLP, Yokohama, Japan (1994) 1043–1046
21. Talkin, D.: A Robust Algorithm for Pitch Tracking (RAPT). In Kleijn, W.B., Paliwal, K.K., eds.: Speech Coding and Synthesis. Elsevier (1995) 495–518
22. Tóth, B.P., Németh, G.: Improvements of Hungarian Hidden Markov Model-based Text-to-Speech Synthesis. Acta Cybernetica **19**(4) (2010) 715–731
23. Imai, S., Sumita, K., Furuichi, C.: Mel Log Spectrum Approximation (MLSA) filter for speech synthesis. Electronics and Communications in Japan (Part I: Communications) **66**(2) (1983) 10–18
24. Olaszy, G.: Precíziós, párhuzamos magyar beszédadatbázis fejlesztése és szolgáltatásai. Beszédkutatás 2013 (2013) 261–270
25. Tóth, B.P., Németh, G., Olaszy, G.: Beszédkorpusz tervezése magyar nyelvű, rejtett Markov-modell alapú szövegfelolvasóhoz. Beszédkutatás 2012 **20** (2012) 278–295
26. Yamagishi, J., Kobayashi, T., Nakano, Y., Ogata, K., Isogai, J.: Analysis of Speaker Adaptation Algorithms for HMM-Based Speech Synthesis and a Constrained SMAPLR Adaptation Algorithm. IEEE Transactions on Audio, Speech, and Language Processing **17**(1) (2009) 66–83
27. Tóth, B.P., Németh, G.: Optimizing HMM Speech Synthesis for Low-Resource Devices. Journal of Advanced Computational Intelligence and Intelligent Informatics **16**(2) (2012) 327–334

Szintaktikai címkekészletek hatása az elemzés eredményességére

Simkó Katalin Ilona[1,2], Kovács Viktória[2], Vincze Veronika[1,3]

[1]Szegedi Tudományegyetem, Informatikai Intézet,
Szeged, Árpád tér 2.
simko@hung.u-szeged.hu
[2]Szegedi Tudományegyetem, Általános Nyelvészeti Tanszék,
Szeged, Egyetem u. 2.
viki921015@hotmail.com
[3]MTA-SZTE Mesterséges Intelligencia Kutatócsoport,
Szeged, Tisza Lajos körút 103.
vinczev@inf.u-szeged.hu

Kivonat Cikkünkben az univerzális dependencia szintaxis címkekészlet változtatásainak a szintaktikai elemzés közvetlen és a szintaxist felhasználó alkalmazások által elért eredmények változására gyakorolt hatását vizsgáljuk három kísérlet keretében. Megvizsgáljuk a határozói-, az alárendelő mellékmondati-, és a funkciócímkék hatását a standard kiértékelési metrikákkal elért eredményekre, a fő, tartalmascímkék helyes felismerésére, valamint egy adott alkalmazás eredményeire.
Kulcsszavak: szintaxis, dependencia, címkekészlet, kiértékelés

1. Bevezetés

A szintaktikai leírások között ma már nem csak elméletben, hanem a számítógépes gyakorlatban is egyre több alternatíva közül választhatunk. Már a magyarra is léteznek konstituens [1], dependencia [2] és LFG [3] nyelvtani számítógépes nyelvészeti leírások, treebankek, ám az egyes elméleti kereteken belül is több különböző reprezentáció érhető el.

Az egyes keretek konkrét reprezentációi között kisebb és nagyobb különbségekkel találkozhatunk: a konstituens nyelvtani keretben készült Szeged Treebank 1.0 verziójának [4] reprezentációjában csak a főnévi csoportok és a tagmondat határok kerültek annotálásra, a 2.0 reprezentációban [1] már melléknévi, határozószói és más frázisok is jelölve vannak. A dependencia nyelvtan keretében eltéréseket láthatunk például a Szeged Dependencia Treebank [2] és a magyar univerzális dependencia treebank [5] között. Címkekészletüket tekintve egyes címkék az egyik reprezentációban elkülönülnek, míg a másikban nem; valamint egyes kategóriák esetén az elemek kötése is eltérő, például a koordináció esetén.

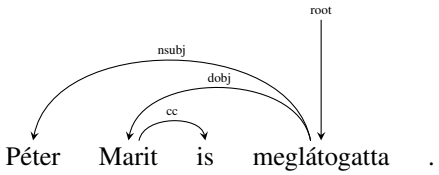
Cikkünkben különböző dependencia címkekészletekkel végzett kísérleteink eredményeit mutatjuk be. Először a standard címkézett (LAS) és címkézetlen (UAS) kiértékeléssel kapcsolatos problémákat mutatjuk be, majd a kísérleteinkhez felhasznált címkekészleteket. Végül közöljük az eredményeinket és egy NLP-s feladat kapcsán bemutatjuk azt is, hogy az eltérő címkekészletek használata szignifikánsan befolyásolja annak eredményességét.

2. Kiértékelés és címkekészletek

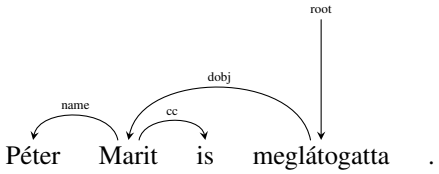
Dependencia szintaktikai elemzések közötti különbségeket általában az elemzők által elért UAS és LAS eredmények alapján állapítunk meg. Ezek a kiértékelési metrikák minden szót egyformán figyelembe vesznek: UAS eredmény esetén a megfelelő helyre kötött szavak, LAS esetén a megfelelő helyre, megfelelő címkével kötött szavak százalékos arányát viszonyíthatjuk egymáshoz.

Egy mondatban egy funkciószó téves kötése ugyanolyan hatással van az UAS és LAS eredményekre, mint egy tartalmas szóé, annak ellenére, hogy mind nyelvészeti szempontból, mind egy alkalmazás számára sokkal "nagyobb hiba" a tartalmas szó tévesztése. Az (1) ábrán a *Péter Marit is meglátogatta* mondat helyes dependencia nyelvtani szerkezetét láthatjuk az univerzális dependencia reprezentációjában. A (2) ábrán az alany hibásan a tárggyal együtt névelemként van elemezve, míg a (3) ábrán az *is* funkciószó hibája látható. Mivel a többi címke és kötés helyes, a (2) és a (3) mondatok UAS és LAS eredményei megegyeznek.

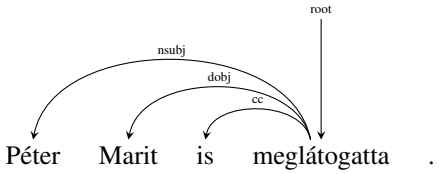
(1)



(2)



(3)



Álláspontunk szerint a szintaktikai elemzés önmagában nem végalkalmazás, hanem az előfeldolgozás része magasabb szintű alkalmazások számára, ezért nem egyforma fontosságú minden nyelvtani szerepű eleme a mondatnak. A standard UAS és LAS kiértékelések ezt nem mindig tükrözik megfelelően. Erre megoldást jelenthet a súlyozott kiértékelés, ahol a fontosabbnak ítélt címkék nagyobb súlyozással, a funkciószavak kisebb súlyozással járulnak az összesített eredményhez; a címkékre kivetített F-mérték, amelyben a számunkra fontosnak ítélt címkék által adott eredményt vehetjük figyelembe; vagy az adatbázisok átcímkézése, ahol a kevésbé releváns címkék összevonásával javítható lehet az elemző releváns címkéken elért teljesítménye. Kísérleteink kiindulópontja a magyar unvierzális dependencia treebank [5], amelyből több treebanket hoztunk létre a teszteléshez különböző címkék összevonásával. Így három típusú, öt darab új treebanket hoztunk létre: a határozószói címkék összevonásával, az alárendelő címkék összevonásával, valamint a funkciószavak címkéinek összevonásával.

Ezeket egymással és az eredeti treebankkel UAS és LAS eredményeken kívül a különböző címkékre mért F-mérték szempontjából hasonlítottunk össze egymással, valamint egy alkalmazásban felhasználva. A következőkben bemutatjuk az egyes új címkekészleteket.

2.1. Határozószók

A magyar univerzális dependencia treebank a Szeged Dependencia Treebank-ből [2] "örökölte" szemantikai információkat is tartalmazó határozói címkéit, amelyek megkülönböztetnek idő és helyhatározókat, és az irányhármasság szerint is különbséget tesznek. Így például a *tto* szintaktikai címke a "meddig" kérdésre válaszoló időhatározót jelöl, míg a *locy* címke "hol" kérdésre válaszoló helyhatározókat kapcsol a szerkezethez. Álláspontunk szerint, ezek között a címkék között dönteni már nem a szintaxis feladata, hanem szemantikai megkülönböztetés.

Két új címkekészletet hoztunk létre: az elsőben időhatározó (*advmod:time*) és helyhatározó (*advmod:loc*) kategóriákká vontuk össze az eredeti 6 címkét, a másodikban mind a hat címkét a már meglévő, általános határozói *advmod* címkével vontuk össze. Kutatási kérdésünk ebben a kísérletben, hogy ezeknek a szemantikai jellegű kategóriáknak az összevonásával nő-e a szintaktikai elemzés hatékonysága.

2.2. Alárendelés

Az univerzális dependencia projektben [6] bevezetett címkekészlet kilenc különböző címkét használ alárendelő mellékmondattípusok megkülönböztetésére. Második kísérletünkben arra voltunk kíváncsiak, hogy milyen hatással van az eredményekre a sokféle alárendelő mellékmondati címke.

Ebben az esetben egy új címkekészletet készítettünk, amelyben ezt a kilenc címkét vontuk egy kategóriába.

2.3. Funkciószavak

Legfőbb célunk a funkciószavak-tartalmas szavak megkülönböztetés vizsgálata volt. Álláspontunk szerint a szintaktikai elemzés legfontosabb célja a fő tartalmas szavak szintaktikai viszonyainak helyes felismerése, így a mondatok állítmányának, alanyának és tárgyának felismerése. Kíváncsiak voltunk, hogy a kisebb funkciócímkék összevonása hogyan változtatja meg a szintaktikai elemzők által elért eredményeket.

Ebben a kísérletben szintén két új címkekészlettel dolgoztunk: az első esetben a legtisztábban funkciócímkéket vontuk egy *funct* címke alá, a második esetben az összes funkciócímkét két új címke alá vontuk össze az erősen funkciócímke típusúakat, és a funkció- és tartalmascímkék között elhelyezhetőek elkülönítve.

Kutatási kérdésünk, hogy a szintaktikai elemzést felhasználó alkalmazások számára kevésbé fontos funkciócímkék összevonása megnöveli-e a szintaktikai elemzés hatékonyságát egészében, UAS és LAS eredményeket tekintve, valamint csak a fő, tartalmas címkék figyelembevételével.

Az 1. táblázatban az új címkekészletek láthatóak.

3. Eredmények

A kísérletekben a magyar univerzális dependencia treebank címkéit a fent említett módokon összevontuk, így az eredeti mellett öt teszt treebankkel kísérleteztünk: TIME-

EREDETI	FUNCT1	FUNCT2	SUB	MODE	TIME-PLACE
acl	acl	**funct2**	**cl**	acl	acl
advcl	advcl	**funct2**	**cl**	advcl	advcl
advmod	**funct1**	**funct1**	advmod	advmod	advmod
advmod:locy	**funct1**	**funct1**	advmod:locy	**advmod**	**advmod:loc**
advmod:mode	**funct1**	**funct1**	advmod:mode	advmod	advmod:mode
advmod:obl	**funct1**	**funct1**	advmod:obl	advmod:obl	advmod:obl
advmod:que	**funct1**	**funct1**	advmod:que	advmod:que	advmod:que
advmod:tfrom	**funct1**	**funct1**	advmod:tfrom	**advmod**	**advmod:time**
advmod:tlocy	**funct1**	**funct1**	advmod:tlocy	**advmod**	**advmod:time**
advmod:to	**funct1**	**funct1**	advmod:to	**advmod**	**advmod:loc**
advmod:tto	**funct1**	**funct1**	advmod:tto	**advmod**	**advmod:time**
amod:att	**funct1**	**funct1**	amod:att	amod:att	amod:att
amod:attlvc	**funct1**	**funct1**	amod:attlvc	amod:attlvc	amod:attlvc
amod:mode	**funct1**	**funct1**	amod:mode	amod:mode	amod:mode
amod:obl	**funct1**	**funct1**	amod:obl	amod:obl	amod:obl
appos	**funct1**	**funct1**	appos	appos	appos
aux	**funct1**	**funct2**	aux	aux	aux
case	**funct1**	**funct1**	case	case	case
cc	**funct1**	**funct1**	cc	cc	cc
ccomp	ccomp	**funct2**	**cl**	ccomp	ccomp
ccomp:dobj	ccomp:obj	**funct2**	**cl**	ccomp:dobj	ccomp:dobj
ccomp:obl	ccomp:obl	**funct2**	**cl**	ccomp:obl	ccomp:obl
ccomp:pred	ccomp:pred	**funct2**	**cl**	ccomp:pred	ccomp:pred
compound	**funct1**	**funct1**	compound	compound	compound
compound:preverb	**funct1**	**funct1**	compound:preverb	compound:preverb	compound:preverb
conj	**funct1**	**funct1**	conj	conj	conj
cop	**funct1**	**funct1**	cop	cop	cop
csubj	csubj	**funct2**	**cl**	csubj	csubj
det	**funct1**	**funct1**	det	det	det
dislocated	**funct1**	**funct1**	dislocated	dislocated	dislocated
dobj	dobj	dobj	dobj	dobj	dobj
dobj:lvc	dobj:lvc	dobj:lvc	dobj:lvc	dobj:lvc	dobj:lvc
goeswith	**funct1**	**funct1**	goeswith	goeswith	goeswith
iobj	iobj	iobj	iobj	iobj	iobj
list	**funct1**	**funct1**	list	list	list
mark	**funct1**	**funct1**	mark	mark	mark
name	**funct1**	**funct1**	name	name	name
neg	**funct1**	**funct2**	neg	neg	neg
nmod	nmod	nmod	nmod	nmod	nmod
nmod:att	nmod:att	nmod:att	nmod:att	nmod:att	nmod:att
nmod:obl	nmod:obl	nmod:obl	nmod:obl	nmod:obl	nmod:obl
nmod:obllvc	nmod:obllvc	nmod:obllvc	nmod:obllvc	nmod:obllvc	nmod:obllvc
nsubj	nsubj	nsubj	nsubj	nsubj	nsubj
nummod	**funct1**	**funct1**	nummod	nummod	nummod
parataxis	**funct1**	**funct2**	**cl**	parataxis	parataxis
punct	**funct1**	**funct1**	punct	punct	punct
remnant	**funct1**	**funct1**	remnant	remnant	remnant
root	root	root	root	root	root
xcomp	**funct1**	**funct2**	**cl**	xcomp	xcomp

1. táblázat. A létrehozott címkekészletek. Az EREDETI-től eltérőek félkövérrel kiemelve.

PLACE (idő- és helyhatározói címkék két címkére összevonása), MODE (idő- és helyhatározói címkék összevonása *mode* címkével), SUB (alárendelő mellékmondati címkék összevonása), FUNCT1 (egyértelmű funkciócímkék összevonása egy kategóriába), FUNCT2 (összes nem tartalmascímke összevonása két kategóriába). A treebankeken tízszeres keresztvalidációval a Bohnet parsert [7] tanítottuk etalon morfológiai címkék használata mellett, a kiértékeléshez UAS, LAS és F-mértéket használtunk.

3.1. UAS, LAS és F-mérték globálisan

Az egyes treebankeken elért LAS, UAS és F-mértékek a 2. táblázatban láthatóak.

	LAS	UAS	F-mérték
EREDETI	81,857	84,357	0,924967
TIME-PLACE	81,317	84,364	0,915494
MODE	81,866	84,055	0,935438
SUB	81,236	84,153	0,914999
FUNCT1	81,766	84,176	0,922665
FUNCT2	82,054	84,05	0,938319

2. táblázat. Különböző címkekészletű treebankeken elért eredmények.

Az EREDETI LAS-hoz képest szignifikáns különbséget csak a FUNCT esetekben és a SUB-nál értünk el (McNemar-teszt, $p < 0{,}05$), az idő- és helyhatározói változtatások által hozott különbségek nem szignifikánsak, ezt az magyarázhatja, hogy ezek a szemantikai címkék nincsenek nagy hatással a szintaxisra. Ám ezekből az eredmények ilyen módon való kiértékeléséből csak azt a (előre is nyilvánvaló) következtetést vonhatjuk le, hogy a címkék eltalálása (mikro F-mérték) jobb kevesebb címke esetén, míg a szavak megfelelő helyre kötése (UAS) legjobban az EREDETI, vagyis a legnagyobb címkekészlettel működik globálisan az összes címkét egyformán figyelembevéve. Fő célunk viszont a különböző címkekészletek fő, tartalmas relációkra való hatásának megvizsgálása volt.

3.2. F-mérték a fő címkékre

A 3. táblázatban az egyes címkekészletekkel elért F-mértékek láthatóak a fő, tartalmas címkékre: *root*, a mondat feje; *nsubj*, a tagmondat alanya; *dobj*, a tárgy; *iobj*, részeshatározó, és *nmod:obl*, egyéb esetű, kötelező főnévi bővítmény.

	root	nsubj	dobj	iobj	nmod:obl	TOTAL
EREDETI	0,867	0,873	0,950	**0,496**	0,923	0,888
TIME-PLACE	*0,858*	0,874	*0,948*	0,443	*0,920*	*0,885*
MODE	0,867	0,874	0,951	0,436	0,924	0,888
SUB	0,867	**0,878**	0,949	0,472	**0,929**	**0,890**
FUNCT1	0,863	0,873	**0,952**	0,44	0,923	0,889
FUNCT2	**0,872**	*0,872*	0,949	*0,409*	0,924	0,888

3. táblázat. Fő címkéken elért F-mérték különböző címkekészleteken. Oszloponként a legmagasabb eredmény félkövérrel, a legalacsonyabb dőlttel.

Az adatokból látható, hogy az EREDETI címkekészletnél az iobj címkén kívül minden esetben jobb eredményeket ér el valamelyik új változat. A részeshatározó a többi címkéhez képest nagyon ritka címke, ami magyarázza eltérő viselkedését. Összességében legalacsonyabb eredményeket a TIME-PLACE címkekészlettel értünk el, ami a legalacsonyabb F-mértéket éri el összességében és három fő címkénél is ez hozza a legkisebb értéket. Legjobbnak a SUB címkekészlet tűnik a fő címkéken történt kiértékelésnél: két címkén és összességében is a legmagasabb F-mértékeket éri el.

3.3. Összevont címkék eredményei

A harmadik elemzésünkben az összevont kategóriák által elért eredményt vizsgáltuk összevonás előtt és után, így például az EREDETI címkekészlet esetén az alárendelő mellékmondatok címkéinek összesített (mikro) F-mértékét a SUB címkekészletben az ezeket összevonó címke F-mértékével. A 4. táblázat az új címkekészletek összevont címkéinek és az EREDETI címkekészlet megfelelő címkéinek összesített F-mértékben mért eredményét mutatja. Az EREDETI és SUB, valamint az EREDETI és FUNCT2 összehasonlításokban szignifikánsan jobb az eredmény az összevont címkék esetén (McNemar-teszt, $p < 0{,}05$).

SUB		FUNCT1		FUNCT2	
EREDETI	SUB	EREDETI	FUNCT1	EREDETI	FUNCT2
0,625	0,814	0,941	0,974	0,944	0,973
				0,708	0,817

4. táblázat. Összevont címkék és megfelelő eredeti címkék F-mértékei.

A finom nyelvészeti megkülönböztetéseken alapuló címkék közötti választás nem egyszerű az elemző számára, így az összevont címkéken szignifikánsan jobb eredményt képes elérni. Álláspontunk szerint ezek a megkülönböztetések legtöbb esetben az alkalmazások szempontjából sem relevánsak, ezért összevonásuk nem okoz problémát, főként ha emellett a tartalmas címkéken elért eredmények is jobbak.

4. Az eltérő címkék hatása az enyhe kognitív zavar felismerésére

Az eltérő címkekészletek gyakorlati hatását megvizsgálandó, egy magasabb rendű nyelvtechnológiai feladatban is kísérleteket végeztünk. Munkacsoportunk korábban létrehozott egy gépi tanuló rendszert, mely a páciensek beszédátirataiból kinyert nyelvi jellemzők alapján osztályozza a kísérleti személyeket aszerint, hogy enyhe kognitív zavarban (EKZ) szenvednek-e vagy sem [8]. A rendszerben használt egyik fontos jellemző a tartalmas és funkciószavak aránya volt a páciens megnyilatkozásában.

Jelen kutatásunk eredményeinek tükrében meg tudtuk vizsgálni, hogy vajon a funkciószavak reprezentációja befolyásolja-e az EKZ felismerésének hatékonyságát. Ennek érdekében újratanítottuk a Bohnet parsert az eredeti reprezentációt tartalmazó treebanken, illetve a FUNCT2 reprezentációt tartalmazó treebanken, majd a kapott modelleket lefuttattuk a páciensek beszédátiratain. Az így kapott kétféle függőségi elemzésből nyertük ki aztán a tartalmas szavak, illetve a funkciószavak arányát, ugyanakkor mást nem változtattunk az eredetileg is használt jellemzőkön.

A kétféle reprezentáció alapján nyert jellemzőtért felhasználva végeztük el kísérleteinket, a Weka [9] szoftver döntési fa (C4.5) algoritmusával [10], követve [8] módszereit. Az eredmények szerint az eredeti reprezentációval 57,14%-os pontosságot, míg a FUNCT2 reprezentációval 69,05%-os pontosságot sikerült elérni, vagyis a módosított reprezentáció szignifikáns hatással bír az eredmények javulására (McNemar-teszt, p = 0,0245). Az EKZ automatikus felismerésében elért kísérleti eredményeink tehát alátámasztják, hogy a megfelelő szintaktikai reprezentáció megválasztása fontos szereppel bírhat a végalkalmazások eredményességére.

5. Összegzés

Cikkünkben különböző dependencia nyelvtani címkekészletekkel végzett kísérleteinket és azok eredményeit mutattuk be. Álláspontunk szerint, mind nyelvészeti, mind NLP-s alkalmazások szempontjából fontosabb a tartalmas címkék helyes felismerése egy szintaktikai elemzésnél, mint a funkciócímkéké. Eredményeink alapján, bizonyos címkecsoportok összevonása javíthatja számunkra fontosabb címkék helyes felismerését, sőt bemutattuk, hogy a reprezentáció módosításával egy végalkalmazás eredményét is szignifikánsan javíthatjuk. Az alkalmazásunk számára megfelelően kiválasztott szintaktikai reprezentáció erősen befolyásolja az alkalmazással elérhető eredményeket.

Hivatkozások

1. Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged TreeBank. In Matousek, V., Mautner, P., Pavelka, T., eds.: Proceedings of the 8th International Conference on Text, Speech and Dialogue, TSD 2005. Lecture Notes in Computer Science, Berlin / Heidelberg, Springer (2005) 123–132
2. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian Dependency Treebank. In: Proceedings of LREC 2010, Valletta, Malta, ELRA (2010)
3. Simkó, K.I., Vincze, V., Farkas, R.: Többszintű szintaktikai reprezentáció kialakítása a Szeged FC Treebankben. In Tanács, A., Varga, V., Vincze, V., eds.: X. Magyar Számítógépes Nyelvészeti Konferencia. (2014) 67–73
4. Csendes, D., Hatvani, C., Alexin, Z., Csirik, J., Gyimóthy, T., Prószéky, G., Váradi, T.: Kézzel annotált magyar nyelvi korpusz: a Szeged Korpusz. In Alexin, Z., Csendes, D., eds.: Magyar Számítógépes Nyelvészeti Konferencia. (2003) 238–245
5. Vincze, V., Farkas, R., Simkó, K.I., Szántó, Zs., Varga, V.: Univerzális dependencia és morfológia magyar nyelvre. In: XII. Magyar Számítógépes Nyelvészeti Konferencia, Szeged (2015) 322–329
6. Nivre, J.: Towards a Universal Grammar for Natural Language Processing. In Gelbukh, A., ed.: Computational Linguistics and Intelligent Text Processing. Springer (2015) 3–16
7. Bohnet, B.: Top accuracy and fast dependency parsing is not a contradiction. In: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010). (2010) 89–97
8. Vincze, V., Gosztolya, G., Tóth, L., Hoffmann, I., Szatlóczki, G., Bánréti, Z., Pákáski, M., Kálmán, J.: Detecting mild cognitive impairment by exploiting linguistic information from transcripts. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Berlin, Germany, Association for Computational Linguistics (2016) 181–187
9. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations **11**(1) (2009) 10–18
10. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA (1993)

Magyar nyelvű szó- és karakterszintű szóbeágyazások

Szántó Zsolt[1], Vincze Veronika[2], Farkas Richárd[1]

[1] Szegedi Tudományegyetem, Informatikai Intézet
Szeged Árpád tér 2.
e-mail:{rfarkas,szantozs}@inf.u-szeged.hu
[2] MTA-SZTE Mesterséges Intelligencia Kutatócsoport
vinczev@inf.u-szeged.hu

Kivonat A szóbeágyazási modellek az egyes szavak párszáz dimenziós folytonos térbe való leképezését adják meg úgy, hogy az egymáshoz hasonló szavak közel kerülnek egymáshoz a beágyazási térben. A szóbeágyazások széles körben használatossá váltak az elmúlt években. Jelen cikkben bemutatunk publikusan elérhető magyar nyelvű szóvektorokat, amelyeket 4,3 milliárd szövegszónyi korpuszból építettünk. Az első modellek (word2vec) a szavakat mint alapegységet dolgozták fel. Az utóbbi években több olyan kiterjesztése is született ezen modelleknek, amelyek karakterszintű információkat is ki tudnak aknázni. Ezek a modellek morfológiailag gazdag nyelveken előnyösebbek lehetnek, mint a pusztán szószintű modellek. A cikkben összehasonlítunk ugyanazon adatbázisból épített szó- és karakterszintű szóbeágyazásokat téma- és véleményosztályozási feladatokon kiértékelve.
Kulcsszavak: Szóbeágyazás, karakterszintű szómodell, osztályozás

1. Bevezetés

A szóbeágyazások alkalmazása az utóbbi években nagyban javította egyes természetesnyelv-feldolgozási alkalmazások hatékonyságát. A szóbeágyazások az egyes szavakat egy jellemzően párszáz dimenziós folytonos térbe képezik le, ahol a hasonló jelentésű szavak egymáshoz közel helyezkednek el. A szóbeágyazások nagy ereje abban rejlik, hogy míg az egyes célfeladatokhoz használható annotált adatbázisok mérete általában erősen korlátozott, addig a szóbeágyazások tanítására óriási méretű annotálatlan szövegeket használhatunk. Ennek következtében pedig a célfeladatunk adatbázisában ismeretlen vagy ritkán látott szóalakokat is képesek vagyunk kezelni. A szóbeágyazók tanítása a szavak közvetlen kontextusára épít, azaz hasonló kontextusban előforduló szavak fognak egymáshoz közel elhelyezkedni. A szavakhoz tartozó kontextus alapján kapott vektortérben mind jelentésbeli, mind morfológiai jellemzők is megjelenhetnek.

Jelen cikkben bemutatunk publikusan elérhető magyar nyelvű szóvektorokat, amelyeket 4,3 milliárd szövegszónyi korpuszból építettünk. Az első szóbeágyazási

modellek a szavakat mint alapegységet dolgozták fel. Az utóbbi években több olyan kiterjesztése is született ezen modelleknek, amelyek karakterszintű információkat is ki tudnak aknázni. Ezek a modellek morfológiailag gazdag nyelveken előnyösebbek lehetnek, mint a pusztán szószintű modellek. A cikkben összehasonlítunk ugyanazon adatbázisból épített szó- és karakterszintű szóbeágyazásokat téma- és véleményosztályozási feladatokon kiértékelve.

2. Szóbeágyazási modellek

A szóbeágyazások egy nyelv szavait egy párszáz dimenziós folytonos térben reprezentálják úgy, hogy a hasonló jelentésű szavak egymáshoz közel helyezkednek el térben. Az első modellek az egyes szavakat mint egységeket kezelték. Ebben a megközelítésben ugyanannak a szótőnek két ragozott alakja pontosan úgy különbözik egymástól, mint egy másik szótő (például a *macska* – *kutya* szópár tagjai pontosan ugyanúgy különböznek egymástól, mint a *macska* – *macskát* szópár esetében, azaz ezeket két különálló egységként kezelték a korai modellek). Az elmúlt években több megoldási javaslat is született arra, hogy ezt a problémát orvosolják. A megoldás minden esetben az, hogy karakterszintű információkra támaszkodva építjük fel a szóbeágyazást. Ezek alkalmazása különösen hasznos lehet a morfológiailag gazdag nyelvek esetén, ahol a ragozás miatt a korábban nem látott szavak aránya magasabb az átlagosnál. A ritka szavak problémájára egy másik lehetséges megoldás a szóalakok lemmatizálása [1], aminek mellékhatása, hogy a szóbeágyazás ismeretlen szövegen történő alkalmazásának előfeltétele lesz a szóbeágyazás tanításához használt lemmatizáló lefuttatása, ezzel szemben a karakteralapú módszerek a szóalak ismeretében képesek meghatározni egy új szó helyét a vektortérben. Ennek következtében a karaktersorozatok vizsgálata lehetőséget ad az elgépelésből fakadó hibák kezelésére is, ami különösen fontos lehet például a közösségi médiából származó szövegek esetén.

2.1. Szószintű modellek

Szóbeágyazások tanítására a két legáltalánosabban használt metódus a CBOW és a skip-gram [2]. Mindkét módszer az egyes szavak környezetét veszi alapul. Míg a CBOW esetén a gépi tanulási feladat a környezet alapján a keresett szó predikálása, addig a skip-gram esetén a kiválasztott szó alapján annak környezetére következtetünk.

Cikkünkben a skip-gram modellt követjük, ezt mutatjuk be röviden. Ebben a modelben két mátrixot tanulunk, a szavak és a környezet beágyazását. Az egyes mondatokban minden szóra legyűjtjük annak környezetében előforduló szavakat és a tanulás célfüggvénye a környezetben előforduló szavak megfigyelésének valószínűsége a középső szó feltevése mellett. A valószínűségek egy log lineáris softmax modellel becsülhetőek. A tanítás után a kontextusmátrixot eldobjuk és a szómátrixot használhatjuk szóbeágyazásnak. Az alacsony számítási igénynek köszönhetően a skip-gram modell hatékonyan alkalmazható nagy adatbázisokra is, milliárd szövegszónyi adatbázis feldolgozható egy nap alatt.

2.2. Karakterszintű modellek

A szavak és azok környezetének vizsgálatánál részletesebb reprezentációt kapunk, amennyiben a szavak vizsgálata mellett a szavakban szereplő karaktersorozatokat is figyelembe vesszük. Jelen cikkben a Facebook kutatói által publikált [3] FastTextet alkalmazzuk, ahol a szavak vektorát kiegészítjük a bennük szereplő karakter 3 és 4 gramokkal. Ezen vektorok felett a skip-gram módszerrel tanítunk szóbeágyazásokat. Ennek köszönhetően ha több karaktersorozatot oszt meg két szó, mint például morfémák vagy elgépelt szavak esetén, akkor azoknak is közel kell lenniük a szavak vektorterében.

3. Publikusan elérhető szóbeágyazások magyarra

A szóbeágyazások készítésénél cél volt, hogy minél nagyobb méretű és változatosabb stílusú és forrású szövegeket használjunk fel. Ehhez jó alapot nyújtott a Magyar Nemzeti Szövegtár 2. változata [4,5]. A MNSz2-ben újsághírek, könyvek, Wikipedia és egyéb szerkesztett szövegek mellett találhatók beszélt és közösségi médiából származó anyagok is. Az MNSz2 mellett a Hunglish [6] magyar-angol párhuzamos korpusz magyar nyelvű szövegeit használtuk fel az ismert magyar nyelvű erőforrások közül. Ezt egészítettük ki az origo.hu-ról és az index.hu-ról származó elektronikus újságcikkekkel. Fontos volt, hogy a korpuszban ne csak szerkesztett szövegek, hanem felhasználók által írt, alacsonyabb minőségű, közösségi médiából származó szövegek is megtalálhatók legyenek, hiszen számos alkalmazás dolgozik ilyen szövegeken és igényli az ilyen szövegeken számolt szóbeágyazásokat. Ehhez a gyakorikerdesek.hu oldaról gyűjtöttünk 1,9 milliárd szónyi kérdést és választ. A korpusz ezeken felül tartalmaz az OpenSubtitle [7] oldalról származó 466 millió token terjedelmű magyar rajongói feliratot, ami beszélt diszkurzusok elemzéséhez jelenthet nagy segítséget. Az egyes szövegek szavakra bontására a magyarlanc [8] beépített tokenizálóját alkalmaztuk. Az 1. táblázat tartalmazza az egyes részkorpuszokban szereplő tokenek számát, beleértve az írásjeleket is.

1. táblázat. Felhasznált részkorpuszok méretei.

Korpusz	Tokenek száma (milliárd)
MNSz2	1,53
Hunglish	0,03
Origo	0,15
Index	0,25
gyakorikerdesek.hu	1,87
OpenSubtitles	0,46
Összesen	4,29

Ezen a korpuszon mind szószintű skip-gram, mind karakterszintű szóbeágyazási modelleket tanítottunk. A szóvektorok publikusan és ingyenesen elérhetőek[3] a rgai.inf.u-szeged.hu/w2v oldalon. Legjobb tudomásunk szerint cikkünket megelőzően csak a Polyglot[4] biztosított publikusan elérhető magyar szóbeágyazást, de annak minősége jóval gyengébb, mint az általunk közzétett beágyazásoké.

3.1. Szóbeágyazások kiértékelése

Az elkészült szóbeágyazások kiértékeléséhez két adatbázist használtunk. Véleményosztályozásra az arukereso.hu oldalról letöltött termékértékeléseket használtunk. Az egyes termékekhez megadható előnyöket és hátrányokat alkalmaztuk pozitív és negatív tanító példaként.

Témaosztályozásra videojáték és sport témájú facebook bejegyzésekből készítettünk adatbázist. Ez szolgálhat közösségi médiából származó szövegek témabesorolásának egy megvalósíthatósági tanulmányaként. Forrásnak az alábbi videojátékokkal foglalkozó Facebook-oldalak publikus posztjait használtuk fel: PC Guru, GameStar (Hungary), 576 KByte és a Gameday Iroda, sport témakörben pedig a Nemzeti Sport Online és a FociHíradó oldalakról gyűjtöttünk publikus bejegyzéseket.

Mindkét esetben tehát bináris dokumentumosztályozási problémát fogalmaztunk meg. A témaosztályozásra használt adatbázis 10000 tanító és 2000 kiértékelő példát tartalmaz, a véleménydetekciós adatbázis 5000 tanító és 1000 kiértékelő dokumentumból áll. Mindkét feladatra a tanító és a kiértékelő adatbázison is az egyes címkék 50-50%-ban fordulnak elő.

4. Eredmények

A két szóbeágyazás kiértékelésére az egyes adatbázisokon a FastText rendszer neuronhálókon [9] alapuló dokumentumosztályozóját alkalmaztuk. Az osztályozó legnagyobb előnye, hogy sebességben vetekszik a hagyományos lineáris modelleket alkalmazó algoritmusok sebességével, viszont hatékonyan képes kihasználni a szóbeágyazásokban rejlő lehetőségeket. Az eredményeket a 2. táblázat tartalmazza.

2. táblázat. Szóbeágyazások pontossága téma- és véleményosztályozásra.

	témaosztályozás	véleménydetekció
FastText	90,1	91,3
FastText + Szószintű skip-gram	89,6	90,7
FastText + Karakterszintű skip-gram	93,5	91,7

[3] Maguk a részkorpuszok nyers szövegei semmilyen formában sem érhetőek el és a szóvektorokból nem lehetséges azok visszafejtése sem.

[4] https://sites.google.com/site/rmyeid/projects/polyglot

A FastText dokumentumosztályozója alapértelmezésként a tanító halmazból tanulja meg az egyes szavak szóbeágyazását, de képes korábban – nagyobb adathalmazon tanított – szóbeágyazások alkalmazására is. A szószintű modellre építő nagymennyiségű adaton tanított szóbeágyazásokkal nem sikerült jobb eredményt elérni a külső erőforrást nem használó modellhez képest. Ezzel a karakterszintű modell alkalmazásával a témaosztályozás esetén 3,5, míg véleménydetekció esetén csekélyebb, 0,4 százalékpontos javulást sikerült elérni. Ennek az lehet az indoka, hogy a karekterszintű modell segítségével lehetőségünk volt a nagymennyiségű szövegben nem található szavak reprezentációjának a becslésére is.

5. Összegzés

Ebben a cikkben ismertettünk magyar nyelvű szóbeágyazási modelleket, amelyeket 4,3 milliárd szövegszónyi korpuszból építettünk. Ezek a modellek szó- és karakterszinten is működnek, ami morfológiailag gazdag nyelveken különösen hasznosnak bizonyul az egy szóhoz tartozó lehetséges szóalakok nagy száma miatt. A létrehozott szóbeágyazásokat téma- és véleményosztályozási feladatokon értékeltük ki. A továbbiakban tervezzük a szóbeágyazások más alkalmazásokban való felhasználását is.

A létrehozott szóvektorok szabadon elérhetők a rgai.inf.u-szeged.hu/w2v oldalon.

Köszönetnyilvánítás

Farkas Richárd kutatásait az MTA Bolyai János ösztöndíja támogatta.

Hivatkozások

1. Siklósi, B., Novák, A.: Beágyazási modellek alkalmazása lexikai kategorizációs feladatokra. In: XII. Magyar Számítógépes Nyelvészeti Konferencia. (2016)
2. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26. Curran Associates, Inc. (2013) 3111–3119
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
4. Oravecz, Cs., Sass, B., Váradi, T.: Mennyiségből minőséget. Nyelvtechnológiai kihívások és tanulságok az MNSz új változatának elkészítésében. In: XI. Magyar Számítógépes Nyelvészeti Konferencia. (2015)
5. Oravecz, Cs., Váradi, T., Sass, B.: The Hungarian Gigaword Corpus. In: LREC. (2014)
6. Varga, D., Halácsy, P., Kornai, A., Nagy, V., Nagy, L., Németh, L., Trón, V.: Parallel corpora for medium density languages. In: Recent Advances in Natural Language Processing IV. Selected papers from RANLP-05. (2007) 247–258
7. Tiedemann, J.: Finding Alternative Translations in a Large Corpus of Movie Subtitles. In: LREC. (2016)

8. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP. (2013) 763–771
9. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016)

Egy vakmerő digitális lexikográfiai kísérlet: a CHDICT nyílt kínai-magyar szótár

Ugray Gábor

https://chdict.zydeo.net/
zydeodict@gmail.com

Kivonat: A CHDICT-tel egy nyílt, közösségileg szerkesztett kínai-magyar szótár indul hamarosan útjára. A cikk az előképek kontextusába helyezve mutatja be a munkát, illetve beszámol a kiinduló lexikai tartalom kiválasztásáról, az alkalmazott szótárfordítási eljárásról és közzététel módjáról. A szerző kísérletként tekint a projektre, melyből kiderülhet, származtatható-e kis nyelvpárokra kielégítő minőségű új szótár az elérhető nyílt forrásokból, és élteképes-e a kollaboratív modell ilyen tartalmakra.

Kulcsszavak: kínai-magyar, szótár, fordítás, nyílt, kollaboratív

1 Bevezetés

A digitális korban a nyelvi közösségek kulturális és gazdasági kibontakozását erősen befolyásolja, hogy mennyi és milyen minőségű digitális nyelvi erőforrást képesek előállítani és karbantartani. Jelen munka is ebben a tágabb összefüggésben értelmezhető: az alábbiakban egy nyílt, az interneten ingyenesen kereshető és bárki által szerkeszthető kétnyelvű szótár létrehozásáról számolok be.

Míg az említett kulturális javak előállítása komoly szellemi munkát igényel és a végtermék forintosítható értéket képvisel, ideális esetben legalább a nyelvi infrastruktúra alapelemeinek mindenki számára egyszerűen és ingyenesen hozzáférhetőnek kell lenniük. Erre ígérnek kézenfekvő megoldást a közösségileg szerkesztett, nyílt tartalmak.

A CHDICT hozzávetőleg 10 ezer szócikkel indul útjára a következő hónapokban, és a szigorú értelemben vett nyelvi tartalmon túl célja egy olyan interaktív online felület kialakítása, amely egyéb nyilvánosan hozzáférhető adattárak beépítésével minőségi javulást jelent a nyomtatott szótárakhoz képest.

A CHDICT létrehozásának csupán két tényező állt útjában. Az első, hogy mindeddig nem létezett ingyenesen felhasználható és szabadon hozzáférhető kínai-magyar szótár, amely kiindulási alapként szolgálhatott volna. A második, hogy a szerző valójában nem tud kínaiul. Utóbbi tényből ered a vállalkozás vakmerő jellege.

2 Előképek

A CHDICT mint nyílt, közösségileg szerkesztett tudástár nem előzmények nélküli. Kézenfekvő a Wikipediára gondolni, de ennél speciálisabb, szorosabban a tárgyhoz kötődő előképekre is támaszkodhattam.

Az első ilyen jellegű kezdeményezés az EDICT[1] japán-angol szótár [2] volt, amely 1991-ben indult útjára, és máig is fejlődik, immár 170 ezernél is több címszót tartalmaz. Az EDICT példáját követve az elmúlt 25 évben számos további szótár is létrejött, amelyek egy-egy kelet-ázsiai nyelvet (japánt vagy kínait) kötnek össze európai nyelvekkel. A teljesség igénye nélkül: 1997 óta fejlődik a kínai-angol CEDICT[2] (114 ezer címszó); 1999 óta a japán-német Wadoku[3] [7] (115 ezer címszó); 2006 óta a kínai-német HanDeDict[4] (150 ezer címszó). Magyar viszonylatban említést érdemel a HunNor norvég-magyar szótár,[5] amely egy maroknyi ember kezdeményezéseként indult a 2000-es évek elején, s mára 40 ezer szócikket tartalmaz.

Több mint két évtized tapasztalatai alapján leszűrhetünk néhány tanulságot ezekből a projektekből. Legtöbbjüknek sikerült elérni, sőt jócskán meghaladni a közepes méretet. Egyikük sem alkalmaz bonyolult formátumot a lexikai tartalom reprezentálására: mind a CEDICT, mind a HanDeDict máig az eredeti EDICT-formátumot követi, amelyben egy sor egy szócikknek felel meg.

A CEDICT története rámutat az átgondolt és explicit licencfeltételek fontosságára. Amikor a szerzői jogot implicit birtokló üzemeltető 2007-ben elérhetetlenné vált, a bizonytalanság az anyag továbbélését is veszélyeztette.[6]

A lenyűgöző „külső” terjedelem, azaz a szócikkek magas száma mellett megfigyelhető, hogy „befelé” a nyílt szótárak nem túl kiterjedtek: kevés nyelvtani és metainformációt közölnek, s a szélsőségesen egyszerű formátum miatt azt is kevéssé normalizált formában teszik. Feltételezésem szerint ez a közösségi szerkesztés velejárója: a legtöbb közreműködő nem rendelkezik nyelvészeti háttérrel.

3 Lexikográfiai eljárás

A nyílt kínai-magyar szótár elindításához először is egy tyúk-tojás problémát kellett feloldanom. Ha nincsenek közreműködők, nincsen közösségileg szerkesztett szótár sem. Ha azonban nincs olyan kiinduló anyag, amely méreténél fogva már értéket képvisel a felhasználók számára, nem lesz közösség sem, ami a szótárat bővítené és továbbfejlesztené.

Az alábbiakban leírt eljárás célja, hogy a rendelkezésre álló források maximális kihasználásával belátható időn belül elérjem a szükséges kiinduló állapotot, méghozzá

[1] http://www.edrdg.org/jmdict/edict_doc.html
[2] https://cc-cedict.org/wiki/
[3] https://www.wadoku.de/
[4] https://handedict.zydeo.net/
[5] http://dict.hunnor.net/
[6] Korabeli, immár nem fellelhető levelezőlisták tartalma alapján.

egész egyszerűen a kiválasztott címszavak CEDICT- és HanDeDict-beli angol és német megfelelőinek magyarra fordításával. A cél kimondottan egy *tökéletlen*, de „elég jó” szótár, ami az évek során szervesen javul és bővül majd.

3.1 Terjedelem

A terjedelem meghatározásához nem indulhattam ki az érett mintaképek méretéből. Az egyik kézenfekvő támpont a Kínai Népköztársaság hivatalos nyelvi szintfelmérőjéhez, a 汉语水平考试-höz (Hànyǔ Shuǐpíng Kǎoshì, HSK) közzétett szólista[7] volt. A legmagasabb szint teljesítéséhez elvárt szókincs 6 ezer szót tesz ki.

E lista tekintetbe vétele mellett szól, hogy a szárazföldi Kínába igyekvő nyelvtanulók mindenképpen a fenti vizsgára készülnek fel, így joggal várják el szótáruktól, hogy tartalmazza az előírt lexikai elemeket. Másrészt okkal lehetnek kétségeink, hogy a lista mennyire felel meg a kortárs nyelvhasználatnak. Számos jel mutat arra, hogy a legkorszerűbb tanulói szótárak kivételével a nyelvi segédanyagok többsége nincs szinkronban a tényleges modern nyelvhasználattal. A német nyelv esetén Tschirner [6] mutatta ki, hogy a 4 ezer szócikk nagyságrendű tanulói szótárak a korpuszokból okadatolható leggyakoribb szavak jelentős hányadát nem tartalmazzák, ellenben számos ritkább lexikai elemet feltüntetnek.

Az empirikus szógyakoriságokat a SUBTLEX-CH korpuszból [3] merítettem. A korpusz kínai filmfeliratokat foglal magában, azaz a kortárs köznyelvről ad képet. A közzétett gyakoriségi listát az teszi különösen értékessé, hogy valóban szavakat tartalmaz, nem írásjegyeket, ami a szóhatárokat nem jelölő kínai írás miatt ritkaságnak számít.

Másik írás tárgyát fogja képezni annak elemzése, hogy mennyire tekinthetjük relevánsnak és teljesnek a HSK-szólistát a SUBTLEX-CH korpusz gyakoriságainak tükrében.

Kézenfekvő viszonyítási pont volt a 10 ezres cél kitűzéséhez Bartos Huba és Hamar Imre kiváló, nyomtatott kínai-magyar szótára [1], amely a kiadó közlése szerint összesen 11.750 bejegyzést tartalmaz. Utólagos megerősítésként találtam Naszódi Mátyás megjegyzésére, miszerint „egy ember belátható idő alatt maximum 10.000 tételből álló szótár készítésére képes”. [5]

A CHDICT kiinduló törzse tehát a HSK-vizsgák 6 ezer szavát tartalmazza, kiegészítve a 4 ezer leggyakoribb szóval, amelyek a vizsgák anyagában nem szerepelnek.

3.2 Források

Eljárásom lényege, hogy a kiválasztott szócikkeket a CEDICT angol, illetve a HanDeDict német megfelelőiből magyarra fordítom. Ennek szerzői jogi szempontból nincs akadálya, mivel a Creative Commons licenc forrásmegjelölés mellett engedélyezi a származtatott anyagok létrehozását.

[7] http://www.hskhsk.com/word-lists.html

A CHDICT kezdeti minőségét a fenti két forrás korlátozza alulról, súlyosbítva a fordításból óhatatlanul adódó torzításokkal. A CEDICT-en és a HanDeDict-en kívül ezért kettős céllal több más forrást is tekintetbe veszek a munka során.

Az első cél a minőség javítása. Az angol jelentések fordításakor a fő problémát az angol nyelv szófaji és szemantikai többértelműsége jelenti. Ezt sajnos csak részben ellensúlyozza a HanDeDict bevonása, mivel ennek sok szócikkét eleve a CEDICT fordításaként állították elő. Egyéb források tekintetbe vételével az angolból eredő többértelműséget igyekszem ellensúlyozni.

A második cél a sebesség. A szótárfordításhoz dedikált eszközt fejlesztettem ki, amely a Google és a Bing fordítómotorokból származó gépi fordítások alapján gépelésgyorsító funkciókat nyújt az emberi fordítás bevitele során.

A szótárfordító alkalmazás a CEDICT-en és a HanDeDict-en túl tartalmazza még a Wikipediából származó cikkek címeit, ha a címszó szerepel önálló cikként, és ahhoz angol, német vagy magyar cikk is társul. Ehhez az előkészítési fázisban a Wikipedia letölthető adatbázis-mentéseit[8] dolgoztam fel gépileg.

Az előkészítés eredménye egy többnyelvű XML-fájl, amely címszavanként tartalmazza az összes eddig ismertetett forrást és azok gépi fordításait.

Másodlagos forrásként támaszkodok az ABC Chinese-English Dictionary-re [4], valamint a Bartos-Hamar-féle kínai-magyar szótárra. Ezeket szerzői jogi okokból nem használom fel módszeresen, de elszigetelt esetekben nagy segítséget jelentenek egyes szavak jelentésének tisztázásában. A MOEDICT kínai értelmező szótárat,[9] amely a tajvani sztenderd mandarint írja le, elsősorban a hagyományos írásjegyekkel és tajvani kiejtéssel kapcsolatos inkonzisztenciák feloldására használom. Avantgárd online „kutatási módszerként" említést érdemel még a Google képkeresési funkciója. Az elegendő szkepszissel kezelt eredmények időnként meglepő információkkal szolgálnak egy-egy szó regiszteréről, képzettársításairól, szerencsés esetben referenséről.

3.3 Munkakörnyezet

A szótárfordításhoz munkaeszközként először egy „polcról levehető", kereskedelmi fordítási környezetet vettem fontolóra. Hamar nyilvánvalóvá vált azonban, hogy itt a hagyományos fordítástól igen eltérő feladatról van szó, és érdemes kifejleszteni egy dedikált, egyszer használatos alkalmazást, amely a munka során a fejlesztési idő többszörösét takarítja meg.

[8] https://en.wikipedia.org/wiki/Wikipedia:Database_download
[9] https://www.moedict.tw/about.html

1. ábra. Képernyőkép a szótárfordításhoz használt munkaeszközről.

A felület explicit elemei elsősorban a hatékonyságot és gyorsaságot szolgálják. Ide tartozik az automatikus kiegészítési funkció a gépi fordításokból kigyűjtött szavak alapján, de a források gyorsan áttekinthető, tipográfiailag tagolt megjelenítése és a könyen navigálható címszólista is.

Implicit összetevő a címszavak sorrendezése. A lista alfabetikus rendezése óhatatlanul monotóniához vezetett volna, de nem előnyös a nyers gyakoriság alapú rendezés sem, mert egész másfajta kihívást jelentenek a gyakori, rövid és rendkívül poliszém kínai szavak, mint a ritkább, hosszabb és egyértelműbb elemek. Túl sok hasonló fejtörő egymás után szintén kedélyrombolú hatású.

Az optimális választás egy kétszintű rendezés volt. Az első rendezési szempont a gyakori szavakat sorolja felülre, viszont minden szó alá besorolja azokat a ritkább lexikai elemeket, amelyeknek a szó prefixe, vagy amelyek a szónak prefixei. Így a listán periodikusan váltakoznak a nagy és kis frekvenciájú szavak, s egymás közelébe kerülnek a jelentésükben összefüggő összetett lexikai elemek is. Mivel pedig a lista végén is szerepelnek gyakori szavak, nem lehetséges a lelkesedés alábbhagytával úgy dönteni, hogy mégiscsak elég lesz 7.500 szó, hiszen a hátralevő anyagban elszórva még rengeteg olyan elem szerepel, amelyről nem mondhatok le.

Az írás pillanatában 7.800 szócikk fordítása áll készen, s a tapasztalatok alapján 100 szócikk feldolgozása körülbelül két munkaórát vesz igénybe. Az eszköz minden egyes szócikkre rögzíti a munkaidőt, így később megvizsgálható, hogy van-e összefüggés az egyes szavak lefordításához szükséges idő és a frekvencia, a szó jelentésszáma, a fordítás minősége stb. között.

4 Az elkészült kiindulópont

4.1 Formátum

A szótár technikailag nem más, mint egy letölthető és szabadon felhasználható szövegfájl. Úgy döntöttem, hogy nem definiálok saját formátumot, vagyis a CHDICT is azt a formátumot használja majd, mint az EDICT, a CEDICT és a HanDeDict.

A változtatás mellett szólt volna, hogy ez az igen egyszerű formátum nem képes jólformáltsági feltételeket biztosítani a szófaji megjelölések, címkézett stílusjegyek, nyelvtani információk (pl. számlálószavak, többszótagos igék belső szerkezete, vonzatok) leírására, illetve nem ad lehetőséget az alternatív kiejtési változatok és írásmódok elegáns jelölésére sem.

Erősebbnek éreztem viszont a hátrányoknál azt a sokatmondó tényt, hogy három különböző, immár a százezres méretet meghaladó, közkedvelt szótár is remekül elboldogul a fenti korlátokkal. A bevett formátum további előnye, hogy megkönnyíti a szótár beépítését az elterjedt offline alkalmazásokba, mint a Pleco vagy a Hanping.

A formátum maga egy pillantásra áttekinthető (a színezés természetesen csak az érthetőséget segíti itt, hiszen nyers szövegről van szó):

```
舉辦 举办 [ju3 ban4] /rendez (eseményt, rendezvényt)/szervez/
```

Anélkül, hogy a fenti szintaxison változtatnék, a CHDICT-ben számos szemantikai kiegészítést teszek. Így például a zárójelezett szövegrészek metainformációnak számítanak, a keresésben nem vesznek részt, és bizonyos helyzetekben zárt címkelistáról kell származniuk.

A bejegyzések között, megjegyzésként jelölt sorokban kiegészítő információk állnak majd a soron következő szócikk státuszáról, korábbi verzióiról, a módosítások időpontjáról és szerzőjéről. Ez a kompatiblitiás megőrzésével eltérés az előképektől, mert a CHDICT adatfájlja így elsőként a teljes változástörténetet magában foglalja.

4.2 Licenc

A CHDICT anyagát Creative Commons licenc alatt teszem közzé. Ez részben a felhasznált anyagok licenceléséből eredő kényszer. Fontosabb azonban, hogy a közösségi licenc lehetővé teszi a tartalom továbbfejlődését abban az esetben is, ha az eredeti fenntartó magára hagyja a projektet. Nem utolsósorban pedig etikai szempont, hogy így a közreműködők azonos feltételek mellett megőrizhetik saját szerzői jogaikat az összes hozzájárulásukra, mivel a verziótörténet is az adat szerves részét képezi.

4.3 Közzététel

A munka elkészültével két végterméket teszek közzé. Az egyik a szótári tartalom, amely letölthető lesz mind a szótár weblapjáról, mind egy automatikusan frissülő

Github-repozitóriumból. A másik az említett weblap maga, amelynek forráskódja már most is elérhető egy Github-repozitóriumban.

A keresési funkciók túlmutatnak a kínai és magyar szavak megtalálásán, és a CHDICT szótári anyagát több más forrással ötvözik. Legfontosabb az automatikus kézírás-felismerés és az egy kattintással elérhető vonássorrend-animációk. Mindkét funkció Shaunak Kishore *Make Me a Hanzi* projektjén[10] alapul, amelyhez csekély mértékben magam is hozzájárultam. Apró részlet a kínai szófrekvenciák tekintetbe vétele a magyar keresési eredmények sorrendezésekor. Ugyanaz a célnyelvi szó gyakran több bejegyzésben is szerepel, amelyek közül célszerű a gyakoribb kínai szavakat előre sorolni, hogy a releváns találatok álljanak a lista elején.

Egyenrangú funkciója a weblapnak a nyilvános szerkesztői felület. Akárcsak a Wikipedia „laptörténet" fülén, a CHDICT weblapján is megtekinthető lesz a szótár összes változása, illetve egy-egy szócikk saját változástörténete.

A szerkesztőfelület is számos, nyelvi adatra épülő kényelmi szolgáltatást tartalmaz majd, így például az egyszerűsített címszó bevitele után automatikusan felkínálja az ismert hagyományos változatot és a pinyin-átiratot, illetve ha a címszó megtalálható a CEDICT-ben vagy a HanDeDict-ben, akkor az ezekben álló szócikket. Az efféle funkciók célja, hogy megkönnyítsék a szerkesztési munkát, ezáltal elősegítsék a szótár bővülését és fejlődését. A nyers szöveges adatformátum sem a keresés során, sem a szerkesztőfelületen nem jelenik meg eredeti formájában.

5 Összegzés és kitekintés

Az írásban bemutattam a CHDICT-en eddig végzett munkát, amelynek eredményeként haramosan egy 10 ezer szócikkes, nyílt, közösségileg szerkesztett kínai-magyar szótár indul útjára. Az intellektuális kihíváson túl leginkább izgalmas kísérletként tekintek a projektre, két kérdésre remélve választ.

Elsősorban: Lehetséges-e az elérhető nyílt forrásokra alapozva, belátható munkabefektetéssel létrehozni egy használható méretű és minőségű kétnyelvű szótárat? Ha igen, úgy a munka útmutatásul szolgálhat más nyelvpárok számára is.

Ami a minőséget illeti, a fenti kérdés megválaszolása kihívást jelent. A szótárfordítás nem bevett gyakorlat, s nem összevethető a gépi fordítás kiértékelésével, de az emberi fordítások minőségbiztosításával sem. Szúrópróbaszerűen kiválasztott szócikkek emberi értékelése, más szótárakkal való összevetése kínálkozik lehetőségként egy későbbi vizsgálat számára. Végeredményben azonban a választ a weblap látógatószáma adja majd meg. A HanDeDict weblapját 60-150 látogató keresi fel naponta, akik 500-2000 lekérdezést hajtanak végre. A CHDICT esetén a beszélők számából kiindulva ennek nagyjából a tizedére számítok.

Miután a szótár weblapja elindul, nagy értéket jelentenek majd a naplózott használati adatok. A gyakori lekérdezések kijelölik a bővítés irányát és a magas prioritással gondozandó szócikkeket is, illetve képet adnak arról, mennyire kielégítő a szótár aktuális terjedelme.

[10] https://skishore.github.io/makemeahanzi/

Jócskán van lehetőség a szótár proaktív fejlesztésére is. Kézenfekvő a magyar tulajdonnevek módszeres bevitele, amelyeket a kínai átiratok kiszámíthatatlansága miatt hasznos szerepeltetni. A fontos nyelvtani információk feltüntetése is további értéket jelent, ám ezeknél egyre kevesebb nyílt forrásra alapozhatunk.

A második kérdés, hogy a kezdeti állapot közzététele után életképes-e a közösségileg szerkesztett modell egy olyan „kicsi" és bizonyos tekintetben speciális nyelvpár esetén, mint a kínai-magyar. Ha igen, úgy átültethető-e vajon a modell más, vélhetően nagyobb impaktfaktorú, eltérő kihívásokat és elvárásokat támasztó nyelvpárokra, mint például az angol-magyar? Erre a válasz megjósolhatatlan.

6 Eddig közzétett anyagok

A CHDICT teaser-oldala: https://chdict.zydeo.net
Az összes forrást ötvöző XML-fájl a kezdeti szótárfordításhoz:
https://chdict.zydeo.net/files/backbone.zip
A webes alkalmazás forráskódja: https://github.com/gugray/ZydeoWeb
Az élő webes alkalmazás, amelyen a HanDeDict kínai-német szótár kereshető (illetve hamarosan szerkeszthető): https://handedict.zydeo.net/
A szótár fordításához használt egyedi alkalmazást szívesen az érdeklődők rendelkezésére bocsájtom.

Hivatkozások

1. Bartos, H., Imre, H.: Kínai-magyar szótár. Balassi Kiadó (1998)
2. Breen, J.W.: Building an Electronic Japanese-English Dictionary. JSAA Conference, Brisbane (1995)
3. Cai, Q., Brysbaert, M.: SUBTLEX-CH: Chinese Word and Character Frequencies Based on Film Subtitles. PLoS ONE 5(6): e10729. doi:10.1371/journal.pone.0010729 (2010)
4. DeFrancis, J., Zhang, Y., Mair, V. (eds.): ABC Chinese-English Comprehensive Dictionary. University of Hawai'i Press (2003)
5. Naszódi, M.: Statisztika megbízhatóság a nyelvészetben. Széljegyzetek egy szótárbővítés ürügyén. In: Tanács, A., Varga, V., Vincze, V. (eds.) XI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 34-45. Szegedi Tudományegyetem, Informatikai Tanszékcsoport, Szeged (2015)
6. Tschirner, E.: Häufigkeitsverteilungen im Deutschen und ihr Einfluss auf den Erwerb des Deutschen als Fremdsprache. In: Marello, Carla a.o. (eds): Atti del XII Congresso Internationale di Lessicografia. Alessandria (2006) 1277-1288.
7. Apel, U.: Ein elektronisches japanisch-deutsches Wörterbuch auf Datenbankbasis – Über das Finden von Wörterbucheinträgen im Computer-Zeitalter. In: Gössmann, Hilaria; Mrugalla, Andreas (eds): 11. Deutschsprachiger Japanologentag in Trier (1999) Bd. II, pp. 627–644.

VII. Laptopos bemutatók

Szinkronizált beszéd- és nyelvultrahang-felvételek a SonoSpeech rendszerrel

Csapó Tamás Gábor[1,2], Deme Andrea[1,3], Gráczi Tekla Etelka[1,4], Markó Alexandra[1,3], Varjasi Gergely[1,3]

[1] MTA-ELTE Lendület Lingvális Artikuláció Kutatócsoport,
[2] Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék
csapot@tmit.bme.hu
[3] Eötvös Loránd Tudományegyetem, Fonetikai Tanszék
{marko.alexandra,deme.andrea}@btk.elte.hu, varjasi.gergely@gmail.com
[4] MTA Nyelvtudományi Intézet, Fonetikai Osztály
graczi.tekla.etelka@nytud.mta.hu

Kivonat:
A jelen ismertetés az MTA–ELTE Lingvális Artikuláció Kutatócsoport ultrahangos vizsgálatainak technikai hátterét, az alkalmazott hardver- és szoftverkörnyezetet, illetőleg a folyó és tervezett kutatásokat mutatja be. A magyar és nemzetközi szakirodalmi előzmények tárgyalása után ismerteti az ultrahangnak mint az artikuláció vizsgálatában alkalmazott eszköznek a sajátosságait, összevetve más kísérleti eszközökkel és módszertanokkal. Kitér a kutatási nehézségekre is, mint például az ultrahangkép beszélőfüggő minősége, a nyelvkontúr manuális és automatikus meghatározása, végül bemutatja a kutatócsoport főbb céljait és terveit, mind az alap-, mind pedig az alkalmazott kutatások területén.

Kulcsszavak: artikuláció, fonetika, beszédtechnológia

1 Bevezetés

Az artikuláció (a beszédképző szervek koordinált mozgása) és az akusztikum (a keletkező beszédjel) kapcsolata az 1700-as évek óta foglalkoztatja a beszédkutatókat [1]. Ahhoz, hogy a beszédképző szervek (pl. hangszalagok, nyelv, ajkak) mozgását vizsgálni tudjuk, speciális eszközökre van szükségünk, mivel a legtöbb ilyen szerv nem látható folyamatosan beszéd közben. Magyar nyelvre eddig kevés olyan artikulációs vizsgálat született, amely dinamikus adatokon (azaz nem csak statikus állóképeken) alapul. Lotz az 1960-as években [2, 3], Szende az 1970-es években [4], majd Bolla az 1980-as években [5, 6] röntgenfilm (ún. röntgenogram/ kinoröntgenografikus vizsgálat) technológiával vizsgálta a magyar beszéd artikulációját. Bolla kutatásaiban az összes magyar magánhangzót és mássalhangzót elemezte: a folyamatos röntgenfelvételekből a vizsgált beszédhangokról öt-öt képet átrajzoltak számítógépre, majd a rajzokat fonetikai szempontból elemezték. A tanulmányokban közlésre adták az összes így keletkezett konfigurációt rajzokon, illetve a toldalékcső méreteit táblázatos formában. Ezek az adatok amellett, hogy segítik a magyar beszédképzés mechaniz-

musainak megismerését és az artikulációs bázis feltárását, akár egy mai modern artikulációs elvű beszédszintetizátorhoz is felhasználhatóak lennének.

Bolla és munkatársai egy későbbi tanulmányban részletesen ismertetik a röntgenogramok készítéséhez használt eszközöket és a felvételek módszertanát [7]. Ebből kiderül, hogy a mikroszámítógépes technikát úgy dolgozták ki, hogy az interlingvális hangtani egybevetésekre is alkalmas legyen. Bolla emellett kísérletezett az ajkak (fotolabiogram) és a szájpadlás (palatogram) vizsgálatával is [8]. Az 1980-as évek röntgenes kísérletei után hosszú ideig nem történtek magyar nyelvű artikulációs kutatások, majd 2008-ban Mády elektromágneses artikulográffal vizsgálta a magyar magánhangzókat normál és gyors beszédben [9]. A magyar magánhangzók vizsgálatára újabb vizsgálat is született, melyben a magánhangzókra a beszédben és az éneklésben az alapfrekvencia függvényében jellemző nyelvkontúrokat, ajakpozíciót és az áll helyzetét (azaz az állkapocs nyitásszögét) elemezték szintén az elektromágneses artikulográfia módszerével [10].

1. táblázat: A nyelv mozgásának vizsgálatára használható technológiák összehasonlítása. EMA = elektromágneses artikulográf. MRI = mágnesrezonancia-képalkotás. PMA = permanens mágneses artikulográf.

Technológia	Előnyök	Hátrányok
Röntgen	kiváló térbeli felbontás	káros az egészségre nyelvkontúr követése szükséges
Ultrahang	jó időbeli és térbeli felbontás elérhető ár	nyelvkontúr követése szükséges csak az ultrahangfejre merőleges nyelvállás látszik jól
EMA	kiváló időbeli felbontás pontonként alacsony mérési hiba	csak pontszintű mérés kábelek befolyásolják a beszélőszervek mozgását
MRI	jó a tér- vagy időbeli felbontás (trade-off)	trade-off a tér- és időbeli felbontás között fekvő helyzet, zajos körülmények nyelvkontúr követése szükséges
PMA	jó időbeli és térbeli felbontás	nem adja meg a nyelv pontos pozícióját, csak a becsült helyzetét

A nemzetközi szakirodalomban is számos példát találhatunk a beszéd közbeni artikuláció vizsgálatára, melyek közül a nyelv mozgásának elemzésére a következő technológiák alkalmasak: röntgen [11], ultrahang [12, 13], elektromágneses artikulográf (EMA) [14], mágnesesrezonancia-képalkotás (MRI) [15, 16] és permanens mágneses artikulográf (PMA) [17, 18]. Az egyes technikák előnyeit és hátrányait az 1. táblázatban hasonlítjuk össze. A hat technológia közül az ultrahang pozitívuma, hogy egyszerűen használható, elérhető árú, valamint nagy felbontású (akár 800 × 600 pixel), és nagy sebességű (akár 100 képkocka / másodperc) felvétel készíthető vele. A jó térbeli felbontás azért fontos, hogy a nyelv alakjáról minél pontosabb képet kapjunk, míg a jó időbeli felbontás ahhoz szükséges, hogy a beszédhangok képzésének gyors változását (pl. zárfelpattanás, koartikuláció) is vizsgálni tudjuk. Az ultrahang hátránya viszont az, hogy a hagyományos beszédkutatási kísérletekhez a rögzített képsorozatból ki kell nyerni a nyelv körvonalát ahhoz, hogy az adatokon további vizsgálatokat lehessen

végezni. Ez elvégezhető manuálisan, ami rendkívül időigényes, vagy automatikus módszerekkel, amelyek viszont ma még nem elég megbízhatóak [19, 20]. Ugyanakkor az ultrahang az egyik legelterjedtebb technológia az artikulációs kutatással foglalkozó beszédkutató laboratóriumokban [21].

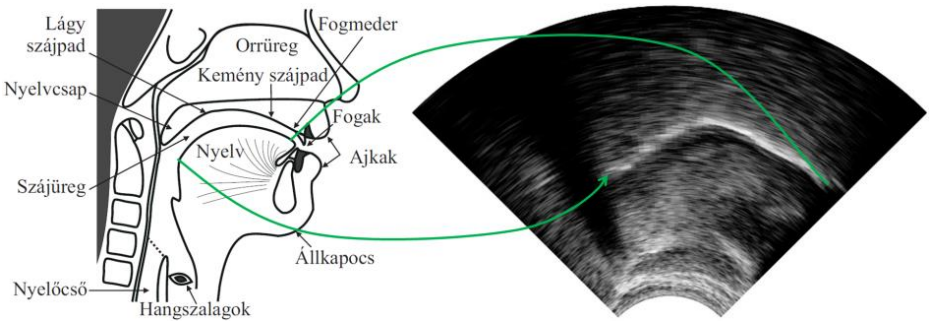
A jelen cikk célja, hogy bemutassuk az MTA–ELTE Lendület Lingvális Artikuláció Kutatócsoport magyar beszéden történő nyelvultrahangos vizsgálatainak technikai hátterét, továbbá azt, hogy a rögzített adatok milyen módon használhatóak fel beszédkutatáshoz.

2 A szinkronizált beszéd és nyelvultrahang felvételének módszertana

Az első kísérleti felvételek az ELTE Fonetikai Tanszékének egyik csendes szobájában készültek, a szakirodalomban javasolt helyzetben és beállításokkal [13], az 1. ábrán látható módon.

A beszélők jelentés nélküli VCVCV szerkezetű hangsorokat és mondatokat olvastak fel. Az ultrahangkép tipikus orientációjára a 2. ábra mutat egy példát.

1. ábra. Beszéd és ultrahang felvétele rögzítő sisakkal az ELTE Fonetikai Tanszéken.



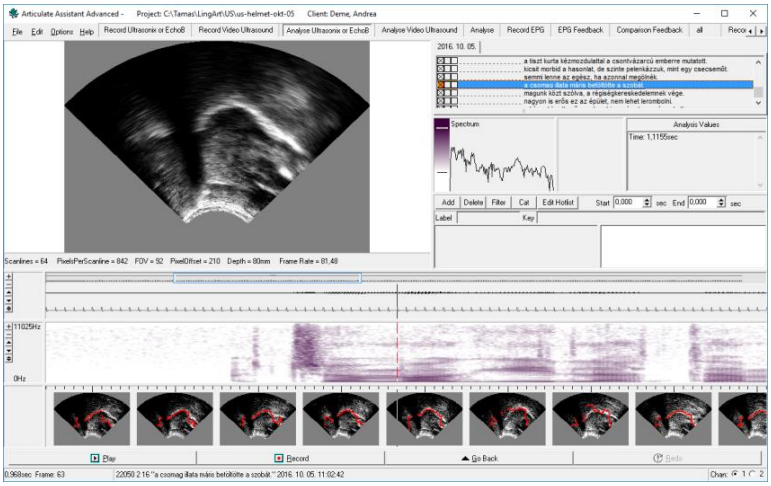
2. ábra. Az ultrahangos kép orientációja. A bal oldali ábra forrása: [22].

2.1 Hardveres környezet

A nyelv középvonalának mozgását a SonoSpeech rendszerrel rögzítettük (Articulate Instruments Ltd.) egy 2–4 MHz frekvenciájú, 64 elemű, 20 mm sugarú konvex ultrahang-vizsgálófejjel, 80–100 fps sebességgel. A felvételek során ultrahangrögzítő sisakot is alkalmaztunk (Articulate Instruments Ltd.), melyet az 1. ábra mutat. A rögzítő sisak használata azt biztosítja, hogy a felvétel során az ultrahang-vizsgálófej ne mozduljon el (pl. az orientációja ne változzon). A beszédet az első kísérletekben Monacor ECM 100 kondenzátormikrofonnal rögzítettük, melyet a kísérleti alany a kezében tartott (az 1. ábrán látható módon). A későbbiekben a beszédet Audio-Technica - ATR 3350 omnidirekcionális kondenzátormikrofonnal rögzítettük, amely a sisakra volt csíptetve, a szájtól kb. 20 cm-re. A hangot 22050 vagy 44100 Hz mintavételi frekvenciával digitalizáltuk M-Audio – MTRACK PLUS hangkártyával. Az ultrahang és a beszéd szinkronizációja a SonoSpech rendszer 'Frame sync' kimenetét használva történt: minden elkészült ultrahangkép után ezen a kimeneten megjelenik egy néhány nanoszekundum nagyságrendű impulzus, amit egy 'Pulse stretch' egység szélesebb négyszögugrássá alakít, hogy digitalizálható legyen (l. 2.3 fejezet). Ez utóbbi jelet szintén a hangkártya rögzítette.

2.2 Szoftveres környezet

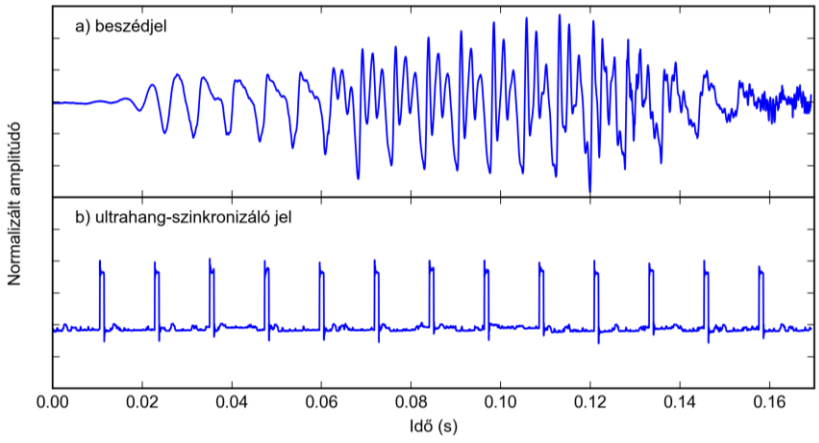
A felolvasandó mondatokat az Articulate Assistant Advanced (Articulate Instruments Ltd.) szoftver segítségével jelenítettük meg a képernyőn. Az adatokat ugyanezzel a szoftverrel rögzítettük. Az AAA szoftver az adatok elemzésére is használható: egyszerre látszik az ultrahangkép, a beszéd hullámformája, FFT-spektruma és spektrogramja (3. ábra). Emellett az ábra alján látható módon automatikus nyelvkontúrkövetésre is alkalmas, és az ultrahangképeket a beszéddel szinkronizáltan jeleníti meg.



3. ábra. Az Articulate Assistant Advanced szoftver használata.

2.3 Beszéd és ultrahang szinkronizálása

Ahhoz, hogy az ultrahangot és a beszédet később együttesen lehessen kezelni (azaz például meg tudjunk nézni egy zárfelpattanáshoz kapcsolódó ultrahangképet), nem elég a két jel párhuzamos felvétele, hanem szinkronizálni is kell azokat. A SonoSpeech ultrahang 'Frame sync' kimenetét (illetve ennek digitalizálható változatát, l. 2.1 fejezet) a kétcsatornás hangkártyára kötve, a mikrofonból származó beszédjelet és az ultrahang-szinkronizáló jelet párhuzamosan fel tudjuk venni (4. ábra). A szinkronizálójelben a négyszögek felfutó élét egy jelfeldolgozási algoritmussal megkeresve meg tudjuk határozni az egyes ultrahangképek pontos helyét a beszédhez képest.



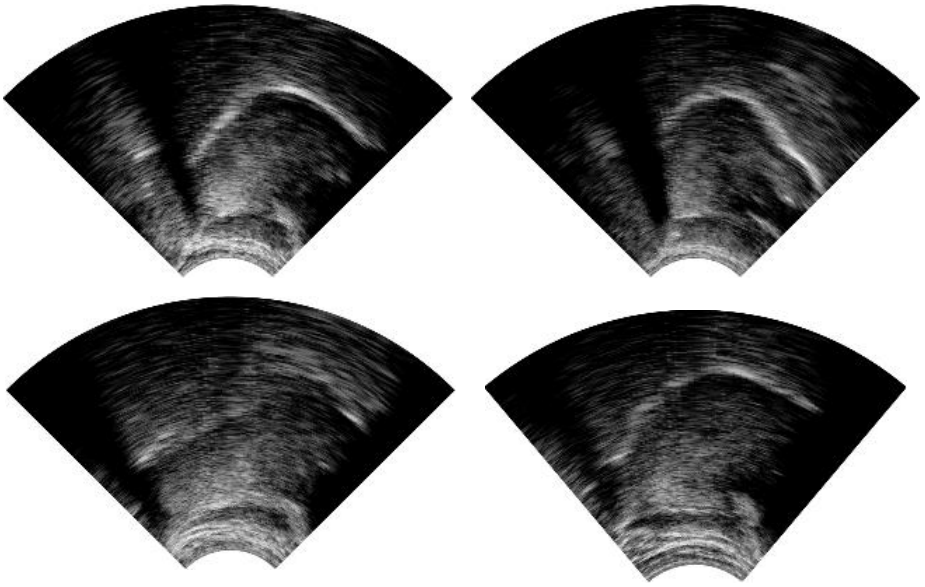
4. ábra. Beszédjel és ultrahang-szinkronizáló jel. Az alsó jelben lévő tüskék az egyes ultrahangképek elkészültét jelölik.

3 Beszéd- és nyelvultrahang-felvételek

3.1 Az ultrahang beszélőfüggősége

Az ultrahangfelvételek képi minősége eltérő lehet az egyes beszélők között. Stone leírja, hogy a fiatal, női, sovány adatközlők artikulációjának ultrahangos rögzítése adja a legjobb képminőséget [13]. Ezt természetesen az artikulációs szervek szöveteinek állapota (pl. hidratáltság) is befolyásolja. Az 5. ábrán négy különböző beszélő azonos beállításokkal készített felvételeiből rögzített képeket láthatunk. A bal oldali sötétebb rész a nyelvcsont helyére, míg a jobb oldali sötétebb rész az állkapocscsont helyére utal (mivel az ultrahang-hullám a csontokon nem tud áthatolni). Látható, hogy a négy különböző beszélő nyelvének felszíne nem egyformán jól látszik. Ennek az is lehet az oka, hogy a rögzítősisak különböző fejméretek esetén máshogy (más orientációban) tartja az ultrahang-vizsgálófejet.

Természetesen a szoftver lehetőséget ad az ultrahangos hardver paramétereinek (pl. vizsgálófej frekvenciája, látómező, mélység, dinamikatartomány, vonalsűrűség stb.) állítására, ez azonban nem minden beszélő esetében kínál elégséges megoldást.



5. ábra. Beszélőnkénti eltérések a nyelvultrahang képeken.
Bal felső: 42 éves nő, jobb felső: 29 éves nő, bal alsó: 31 éves férfi, jobb alsó: 33 éves nő.

A képek minősége befolyásolja a nyelvkontúrkövetéshez használt szoftver teljesítményét is. Automatikus követésre maga az AAA szoftver is kínál lehetőséget, emellett számos más program is rendelkezésre áll. Ilyen például az EdgeTrak [23], a ToungeTrack [24], és az AutoTrace [25] szoftver; illetve a legújabb nyelvkontúrkövető módszerek is használhatóak [26]. Ezen szoftverek között eltér

például, hogy igényelnek-e előzetes manuális betanító adatbázist, avagy képi hasonlóságon alapulnak (összehasonlítás: [19]).

Mindebből következik, hogy a kutatások megtervezésének egyik elengedhetetlen lépése a megfelelő beállítások és a precíz nyelvkontúrkövető módszerek feltárása.

4 Laptopos bemutató

Demonstrációnkban 5 magyar anyanyelvű beszélővel készült ultrahangvideók alapján mutatjuk be az ultrahangos nyelvkontúrkövetés módszereit. A bemutatóban szerepel az ultrahangos mérések képpé alakításának folyamata, specifikációi, a felvételi körülmények ismertetése. Ezután az AAA szoftver működésére térünk rá, különös tekintettel a beállítási lehetőségekre. Emellett a nyelvkontúr követésének korábban említett lehetőségeit ismertetjük, azok előnyeinek, hátrányainak és korlátainak bemutatásával.

5 Kutatási tervek

Megkezdett és jövőbeni kutatásaink során egyrészt a koartikulációnak a nyelvmozgásban detektálható mintázatait elemezzük, másrészt az ultrahangos artikulációkövetésnek a képfeldolgozási és beszédtechnológiai alkalmazásban rejlő lehetőségeit kívánjuk feltárni.

Megindultak kutatásaink az artikuláció ultrahangos képi feldolgozási lehetőségei [27] és az artikuláció alapján történő akusztikumbecslés terén [28], illetve megkezdtük az artikulációs tempó magánhangzóejtésre és ennek kontextusfüggő jellemzőire gyakorolt hatásának feltárását.

Bibliográfia

1. Kempelen, F.: Az emberi beszéd mechanizmusa, valamint a szerző beszélőgépének leírása. Szépirodalmi Könyvkiadó, Budapest. (1989). [Eredeti cím: Mechanismus der menschlichen Sprache nebst der Beschreibung seiner sprechenden Maschine. (1791)].
2. Lotz, J.: Egy magyar röntgen-hangosfilm és néhány fonológiai kérdés. Magyar Nyelv. 62, 257–266 (1966).
3. Lotz, J.: Hangos röntgenfilm-vetítés a magyar nyelv hangképzéséről. Nyelvtudományi Értekezések. 58, 255–258 (1967).
4. Szende, T.: A magyar hangrendszer néhány összefüggése röntgenográfiai vizsgálatok tükrében. Magyar Nyelv. 70, 68–77 (1974).
5. Bolla, K.: A magyar magánhangzók és rövid mássalhangzók képzési sajátságainak dinamikus kinoröntgenográfiai elemzése. Magyar Fonetikai Füzetek. 8, 5–62 (1981).
6. Bolla, K.: A magyar hosszú mássalhangzók képzése. (Kinoröntgenografikus vizsgálat számítógépppel). Magyar Fonetikai Füzetek. 8, 7–55 (1981).
7. Bolla, K., Földi, É., Kincses, G.: A toldalékcső artikulációs folyamatainak számítógépes vizsgálata. Magyar Fonetikai Füzetek. 15, 155–165 (1985).
8. Bolla, K.: Magyar fonetikai atlasz. A szegmentális hangszerkezet elemei. Nemzeti Tankönyvkiadó, Budapest (1995).

9. Mády, K.: Magyar magánhangzók vizsgálata elektromágneses artikulográffal normál és gyors beszédben. Beszédkutatás 2008. 52–66 (2008).
10. Deme, A., Greisbach, R., Markó, A., Meier, M., Bartók, M., Jankovics, J., Weidl, Z.: Tongue and jaw movements in high-pitched soprano singing: A case study. Beszédkutatás 2016 [Speech Research 2016]. 24, 121–138 (2016).
11. Öhman, S., Stevens, K.: Cineradiographic studies of speech: procedures and objectives. J. Acoust. Soc. Am. 35, 1889 (1963).
12. Stone, M., Sonies, B., Shawker, T., Weiss, G., Nadel, L.: Analysis of real-time ultrasound images of tongue configuration using a grid-digitizing system. J. Phon. 11, 207–218 (1983).
13. Stone, M.: A guide to analysing tongue motion from ultrasound images. Clin. Linguist. Phon. 19, 455–501 (2005).
14. Schönle, P.W., Gräbe, K., Wenig, P., Höhne, J., Schrader, J., Conrad, B.: Electromagnetic articulography: use of alternating magnetic fields for tracking movements of multiple points inside and outside the vocal tract. Brain Lang. 31, 26–35 (1987).
15. Baer, T., Gore, J., Gracco, L., Nye, P.: Analysis of vocal tract shape and dimensions using magnetic resonance imaging: Vowels. J. Acoust. Soc. Am. 90, 799–828 (1991).
16. Woo, J., Murano, E.Z., Stone, M., Prince, J.L.: Reconstruction of high-resolution tongue volumes from MRI. IEEE Trans. Biomed. Eng. 59, 3511–3524 (2012).
17. Cheah, L.A., Bai, J., Gonzalez, J.A., Ell, S.R., Gilbert, J.M., Moore, R.K., Green, P.D.: A user-centric design of permanent magnetic articulography based assistive speech technology. In: Proc. BioSignals. pp. 109–116 (2015).
18. Gonzalez, J.A., Moore, R.K., Gilbert, J.M., Cheah, L.A., Ell, S., Bai, J.: A silent speech system based on permanent magnet articulography and direct synthesis. Comput. Speech Lang. 39, 67–87 (2016).
19. Csapó, T.G., Lulich, S.M.: Error analysis of extracted tongue contours from 2D ultrasound images. In: Proc. Interspeech. pp. 2157–2161. , Dresden, Germany (2015).
20. Csapó, T.G., Csopor, D.: Ultrahangos nyelvkontúr követés automatikusan: a mély neuronhálókon alapuló AutoTrace eljárás vizsgálata. Beszédkutatás 2015. 177–187 (2015).
21. Wrench, A.: Ultrasound speech analysis: State of the art. In: Ultrafest VI. , Edinburgh, UK (2013). http://materials.articulateinstruments.com/Technical/State_of_Art.ppt
22. Németh, G., Olaszy, G. eds: A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Akadémiai Kiadó, Budapest (2010).
23. Li, M., Kambhamettu, C., Stone, M.: Automatic contour tracking in ultrasound images. Clin. Linguist. Phon. 19, 545–554 (2005).
24. Tang, L., Bressmann, T., Hamarneh, G.: Tongue contour tracking in dynamic ultrasound via higher-order MRFs and efficient fusion moves. Med. Image Anal. 16, 1503–1520 (2012).
25. Hahn-powell, G. V., Archangeli, D., Berry, J., Fasel, I.: AutoTrace: An automatic system for tracing tongue contours. J. Acoust. Soc. Am. 136, 2104 (2014).
26. Xu, K., Gábor Csapó, T., Roussel, P., Denby, B.: A comparative study on the contour tracking algorithms in ultrasound tongue images with automatic re-initialization. J. Acoust. Soc. Am. 139, EL154-EL160 (2016).
27. Xu, K., Roussel, P., Csapó, T.G., Denby, B.: Convolutional neural network-based automatic classification of midsagittal tongue gestures using B-mode ultrasound images. submitted to J. Acoust. Soc. Am. Express Lett., (2016).
28. Csapó, T.G., Grósz, T., Tóth, L., Markó, A.: Beszédszintézis ultrahangos artikulációs felvételekből mély neuronhálók segítségével. In: Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2017), Szeged, Magyarország, (2017).

A magyar helyesírás-ellenőrzők mai állása

Naszódi Mátyás, e-mail: naszodim@morphologic.hu

MorphoLogic, 1122 Ráth György utca 36.

Kivonat A helyesírás-ellenőrzők jósága függ az előállítás módjától, karbantartásától, de az adatbázis méretének növekedésével objektív korlátokba ütközik a minőség. Jelen cikk kitér az objektív minősítés módszertanára, elvi korlátaira. Összeveti az elérhető helyesírás-ellenőrzőket. Megkísérli pártatlan módon összevetni az elérhető programokat, és megmutatni, hogy a nyelvi adatbázis építésénél alkalmazott módszereknek milyen előnyük, hátrányuk van. A cikk végén keresi a további hatékony fejlesztés irányát.

Kulcsszavak: szóellenőrzés, statisztika, nyelvminőség

1. Bevezető

A helyesírás-ellenőrzők a személyi számítógépek megjelenésével terjedtek el. Angol, majd francia, spanyol, olasz nyelveken íróknak könnyítette meg a dolgát. Magyarra készített szpellerek a 90-es évek elején jelentek meg. A késést nyelvünk összetettsége okozta. Míg az indoeurópai nyelveknél elegendő pár százezer szóalakot azonosítani egy gépi lektornak, addig magyar, finn, török nyelveknél az eszköznek több milliárd alakot kell felismernie.
Mostanában jelent meg a palettán a Microsoft és a Google ellenőrzője. Nyelvtanunk hivatalosan is megújult, melyet az eszközöknek is követnie kell.
Jelentek meg helyesírás-ellenőrzők tesztjéről szóló cikkek[1][2], de ha tesztanyag az eszköz előállításánál szerepet játszott, akkor arra az eszközre aránytalanul jó eredményhez vezet.

2. Technikai áttekintés

A 80-as években olvastam egy írást azzal a címmel: Hogyan készítsünk helyesírás-ellenőrzőt?. A recept a következő: egy szótárba gyűjtsük az ismeretlen szavakat. Ha a program találkozik egy új szóval a szövegben, a felhasználó döntsön, kell-e. A szavak gyakorisági statisztikája miatt a szöveg felét a szóalakok kis hányada lefedi – akár ezer szó – a módszer az angolra be is válik. A magyarra az ilyen próbálkozás teljes kudarcba fulladt.
Ragozó, agglutináló nyelvekben túl sok szóalak létezik. Nem lehet összegyűjteni annyit, hogy ezekkel elfogadható lefedettséget érjünk el. 2016-os cikkemben[3] említem, hogy exponenciálisan csökkenő valószínűséggel előforduló egyedek gyűjtése megfelelő hibaszázalékkal csak korlátos mennyiségben lehetséges. Nyelvi adatbázisok építésénél ez maximum 200 000 körüli érték. Hasonló adatokat említ

Kornai András *Frequency in morphology*[4] című írásában.
Szavakat kell gyűjteni, és generatív modell alapján kell előállítani a szóalakokat – vagy a nyelvi leírás alapján kell visszavezetni a szóalakot morfémák sorozatára. Olyan megoldásokat, melyekkel a szóellenőrzők nyelvi adatbázisainak mérete a kritikus alá csökkenhetett, csak a 80-as évek végétől készítettek.
A generatív modell miatt a szóalakok nem feltétlen gyakoriságuk miatt kerülnek be a készletbe. Ha egy szót regisztrálunk, akkor annak minden szabályosan toldalékolt alakját is, még ha nem is használatosak. Ezek olyan közel lehetnek egy gyakori szóalakhoz, hogy nagy az esélye, hogy a helyes szó elütése következtében került a papírra. A *tan* főnév *-i* képzős alakja tárgyesetben *tanit*, amit gyakran írnak le a *tanít* helyett. A magyar nyelv nagyon sűrű, különböző szavak nagyon közel vannak egymáshoz. Emiatt magyar nyelvnél az előbb említett probléma gyakrabban merül fel, mint angolban, németben, olaszban...

3. A választék

Jelenleg a következő általánosan használható helyesírás-ellenőrzők léteznek:

- Helyes-e?: A MorphoLogic terméke. MS Office-ok része volt. Sok más alkalmazásba került bele. Megjelenése: 1992. Alkotói: Prószéky Gábor, Pál Miklós, Tihanyi László. Jelen fejlesztők közül kiemelném Novák Attilát.
- Lektor: Seregy Lajos nyelvész és a MicroSec programozóinak terméke. Elsőnek, még a 80-as évek végén jelentették be, de végül 1992-ben lett belőle eszköz. Sajnos azóta nem fejlődött.
- Helyeske: Elekfi László ragozási paradigmaszótárára épülő véges automata elven működő ellenőrzőt Farkas Ernővel készítettem. 1993-ban lett a MorphoLogic terméke, de azóta nem fejlődött tovább.
- ISPELL, MYSPELL és HUNSPELL: a szabad szoftverek világában fejlődő vonal. Két szempontból is jelentős. Egyrészt a HUNSPELL, a legfejlettebb változat magyar gyártmány. Szabad szoftver lévén sok helyen használják böngészőnél, levelezőnél. Legmarkánsabb javulását a Szószablya[5] keretében végezték rajta. Alkotója Németh László. A nyelvi leírásnak számtalan „bedolgozója" volt.
- Kimmo-féle kétszintű morfológia: a XEROX-nál, IBM-nél használják. Ezek magyar nyelvi kiindulási anyagát a MorphoLogic állította elő, de nem helyesírás-ellenőrző céljából, és azóta sokat változott.
- A Microsoft ellenőrzője: Egyetlen program kezeli a különböző nyelveken írt szövegek javítását. A Microsoft ellenőrzője 2015 óta működik. Az új MS Office-ok szerves része, emiatt sokaknak lesz hozzá szerencséje.
- Hozzáférhetők webes felületű helyesírási tanácsadók[6][7][8]. Ezeket három okból nem vettem górcső alá.
  1. Tömegfelhasználásban kevésbé játszanak szerepet.
  2. Nehéz a 2-es típusú hibát detektálni (lásd később)
  3. Nem lehet vele nagy tömegű anyagot tesztelni.

A Helyes-e, Helyeske, HUNSPELL forrásai számomra hozzáférhetőek, ezért minősítéseimet megalapozottak, míg a Microsoft forrásanyagára csak a viselkedés alapján következtethetek.

4. Mennyiségi teszt lektorálatlan szövegen

Kétfajta tévedés lehetséges.

1. Helyes szót nem ismer fel, tehát hibásnak tart.
2. Helytelen szót helyesnek minősít, ezért elfogadja

Ha a szövegszerkesztőben az 1-es típusú tévedés fordul elő, a program jelez. A második esetben a felhasználónak nem jut tudomására a szöveghiba, emiatt a szöveg javítatlan marad. Kiss G Gábor cikkében[9] 10-szeres súllyal bünteti a 2-es hibát. A fent vázolt gondok miatt ennél jóval nagyobb a jelentősége.

4.1. Elvi megfontolások

Az, hogy egy karakterlánc magyar szó-e, valószínűségi kérdés. Hibásnak ítélt szó is lehet helyes: *nemecsek*, *frisssssss*, de a *böszmeség* is csak azóta ismert, mióta kiszivárgott az öszödi beszéd. A szövegekben előforduló sztringek többségéről minden magyar anyanyelvű határozottan tud dönteni. Ennek az oka, hogy a valószínűségek elég karakterisztikusak. A többség vagy megüt egy szükséges szintet, vagy egy nagyon alacsony szint alatta marad. A kettő közötti hányad, mely esetekben esetleg még nyelvészek sem értenek egyet, elenyésző.
A nagy valószínűségű szavaknál a statisztikai becslés megbízhatósága elfogadható, de a szavak többségénél, még ha megütik az elfogadható szintet, a statisztikai becslés megbízhatósága alacsony.

1. A szóalakok előfordulási valószínűsége szövegkörnyezettől függ.
2. A szóalakok előfordulási valószínűsége írótól függ.
3. A szóalakok előfordulási valószínűségét csak a gyakoribb esetekben lehet megbízhatóan becsülni.
4. Ha lenne is megbízható becslés, ennek felhasználása a mai számítástechnika mellett túl nagy erőforrást igényelne.
5. A felhasználót irritálná, ha a szavakról a program nem jó-rossz választ adna. Még a „talán” válasszal sem tudna mit kezdeni.

Mindezek miatt a nyelvi adatbázisok és az erre épülő programok igen-nem döntést hoznak a szóalakokról, melynek egy küszöbszint elérése lehet az alapja.

4.2. Technikai meggondolások

Hogy egy szót elfogad-e vagy sem a program, a futtatás választ ad. Arra a kérdésre viszont, hogy helyes-e a szó, nincs objektív mérce. Vagy nagy kompetenciával rendelkező emberi erőforrást kell igénybe vennünk, vagy le kell mondanunk a szavak egyedi minősítéséről. Mivel a teszt során feldolgozandó anyag mérete tetemes, az emberi minősítés nem jöhet szóba.
Az eszközök összevetéséhez nem kell vizsgálni azokat a szavakat, melyekről mindegyik azonosan dönt. A relatív minősítésben csak az eltérések játszanak szerepet.

A jelen vizsgálatnál az eltérően bírált szavak száma 1000-es, 10 000-es nagyságú. Az egyszerű előfordulási statisztika nem segít, mert számos, mindenki által elfogadott szóalak létezik, mely egyszer sem volt leírva. (Valószínűleg az a szó, hogy *testetlenítettséggel* most lett először leírva, de helyes szó.) Egyes hibás alak előfordulási gyakorisága ezt jóval meghaladja. (*Hüje, írts, szervíz...*)
Ha csupán két ellenőrzőt vetünk össze, akkor kikeressük azokat a szavakat, melyeknél ellentétes döntés született. A legegyszerűbb kiértékelés, ha a hibás döntések számát vetjük össze. Amelyiknél kisebb ez az érték, az lehet a jobb ellenőrző. Ennél egy fokkal jobb, ha a 2-es típusnak nagyobb súlyt adunk.
Ha ismernénk a szavak valószínűségét, súlyozhatnánk vele. Egy gyakori szó elhibázása nagyobb baj, mint egy ritkáé. Hát még egy gyakori hiba megengedése. A legpontosabb minősítés az lenne, ha azt is felismernénk, mekkora kárt jelent egy ilyen téves szó. Vagyis a globális képlet:

$$\sum W(alak) = \sum_{alak} p(alak) * e(alak), \tag{1}$$

ahol W a hiba súlya, p az alak valószínűsége, e pedig a hiba által okozott kár, tehát a helyes szóalakoknál $e(alak) = 0$.
Ha a kárt abban mérjük, hogy milyen szóalakok elírásából adódhatnak, a következő becslést adhatjuk.

$$\sum_{alak} p(alak) * e(alak) = \sum_{alak} p(alak) * \sum_{alak2 \in Helyes} \frac{p(alak2)}{e^{m(alak,alak2)}} \tag{2}$$

ahol m a két szóalak közti távolság, amit már mérni, számolni lehet[3].
Ha nincs a valószínűségre sem jó becslés, akkor egyszerűbb képletet kell alkalmazni. A korábbiak szerint a valószínűség becslése csak a gyakran előforduló szavaknál lehetséges.

4.3. A tesztkorpusz

A Népszabadság 1993-as szerkesztősége rendelkezésünkre bocsátott egy nagyobb mennyiségű anyagot. Egyéb forrásunk nagyobb hányada a magyar szpellerek megszületését megelőző időkből származik.
A tesztkorpusz mérete 5 585 000 karakter, 745 900 szó 131 000 különböző szóalak. A 30 leggyakoribb szóalak lefedi a szöveg 25 %-át. Az első 15 alak:

a	54394		*hogy*	11215		*volt*	2356		*vagy*	2141		*kell*	1579		*el*	1427
az	20280		*A*	9789		*de*	2277		*s*	2059		*szerint*	1533		*ki*	1356
és	13520		*nem*	8658		*már*	2167		*még*	2054		*van*	1494		*mert*	1265

A ritkán előforduló szóalakok számából látszik, hogy a többség csak egyszer fordul elő:

1-szer fordul elő 77820 szóalak	2-szer fordul elő 21351 szóalak
3-szor fordul elő 8604 szóalak	4-szer fordul elő 5085 szóalak
5-ször fordul elő 3299 szóalak	6-szor fordul elő 2261 szóalak
7-szer fordul elő 1699 szóalak	8-szor fordul elő 1272 szóalak

A 131 000 szóalakból az ellenőrzők más-más szavakat tartottak hibásnak:

Office 6	Office XP	Office 2002	Office 2016	HUMOR 97	HUMOR 2000
15500	12900	12000	15500	11000	16000
ISPELL	MYSPELL	HUNSPELL	Libre Office	Lektor	Helyeske
17500	17900	13300	13100	17000	20300

A táblázat a 2-es típusú hiba becslésére nem ad lehetőséget. Vizsgáljuk meg, melyek azok a szavak, melyeket az egyik ellenőrző elfogad, a másik elutasít.

	Off 6	OXP	2002	2016	O 97	H 97	2000	ISP	MYS	HUN	Lekt	Heke
Office 6		4166	3516	6258	3129	4897	1402	5113	2615	3449	2291	1527
Office XP	1552		706	3980	861	3027	1996	3828	1565	2721	2129	1968
Office2002	926	730		3664	206	2485	1390	3940	1716	2165	1920	1364
Office2016	2794	3130	2790		2872	3569	2925	3341	2997	3467	2436	2929
Office 97	750	1096	416	3958		2435	1253	4181	2033	2126	2027	1295
HUMOR97	399	1143	577	2535	316		126	2960	991	730	833	481
HUM2000	1918	5126	4496	6905	4148	5139		5541	3140	4008	3103	2414
ISPELL	4758	6086	6144	6449	6204	7103	4672		1344	5309	3666	4672
MYSPELL	4628	6192	6312	8474	4625	7501	4637	3743		5274	4314	3847
HUNSPELL	1252	3138	2558	4733	2307	3030	1294	3468	1063		1599	1062
Lektor	3837	6988	6055	7744	5951	6876	4132	5567	3846	5342		4062
Helyeske	6352	9407	8779	11218	8500	9804	6722	9184	6659	8085	7342	

Ha a szóalakok előfordulási gyakoriságát is figyelembe venném, a fenti teszt nem mutatna ki még ilyen kis különbséget sem. A lefedettség mindegyiknél 97 % körüli érték. Szubjektív módon érzi a felhasználó, hogy melyik a jobb, de ezt nehéz így számszerű adattal igazolni. Emiatt finomabb különbségtételre van szükség.

5. Teszt mesterséges tesztanyaggal

A magyar ABC kisbetűiből álló legfeljebb 6 karakteres sztringeket ellenőriztem egy ponttal lezárva. Majd 2 200 000 000 szóalak keletkezik. A szó végi pontot mindegyik program aszerint kezelte, hogy kötelező vagy nem a szó után.

2 176 782 336	Office XP	Office 2002	Office 2016	HUNSPELL	Helyeske
futási idő	6 óra	3 nap	10 nap	30 perc	1 perc
helyes szavak	600 037	594 409	3 910 312	776 515	281 511
ebből ponttal a végén	80	101	1 298 036	290	68

Ezek az adatok még markánsabban mutatják a különbségeket. A HUNSPELL-nél azért magasabb a ponttal végződők száma, mert a római számokat csak ponttal lezárva fogadja el. Az Office 2016-nál az a hiba állt elő, hogy rövidítéseket is megenged szóösszetételben. Ez okozza a mérhetetlen nagy számot.
Az eggyel felette levő sorban az Office 2016 imponáló adata onnan ered, hogy rengeteg hibás szóalakot fogad el: sok kötőjellel toldalékolandó szót kötőjel nélkül. Ráadásul ezeket szóösszetételben is használja. Ilyen mellélövések mellett az egyéb hibák száma eltörpül.
Az Office 2016 hétkarakteres szavaknál 1 évig futott volna! A Helyeske imponáló ideje lenyűgöző akkor is, ha a tesztágy különböző volt. A sebesség egy szövegszerkesztőnél nem lényeges. A szöveg beírása jóval lassabb ennél. Azt is figyelembe lehet venni, hogy az algoritmusok helyes szavaknál sokkal hatékonyabbak, mint hibás szó esetén, és ez utóbbi tesztnél szinte mindegyik szó hibás volt.
Érdemes a kereszteszt adatait is megtekinteni, hisz ebből már olyan adathalmazok keletkeznek, melyeket közvetlen emberi erővel nem, de mintavételezés után érdemes lenne vizsgálni.

	Office XP	Office 2002	Office 2016	HUNSPELL	Helyeske
Office XP		48 936	3 440 268	333 014	98 563
Office 2002	54 564		3 446 664	327 906	95 098
Office 2016	129 992	130 761		168 872	105 112
HUNSPELL	156 536	145 800	3 320 668		109 458
Helyeske	417 089	407 996	3 733 913	604 462	

Az adatok most is az Office 2016 oszlopában a legnagyobbak. Ha belenéz valaki az állományokba, kiderül az oka. Több mint 3 000 000 hibásan elfogadott szó. A becslés onnan ered, hogy véletlenszerűen kiválasztva az elfogadott szavakból egy részhalmazt, annak legalább három negyede helytelen forma.
Utólag még ráengedtem ezt az iromány t és a Tinta kiadó helyesírási szótárát is az ellenőrzőkre. A tanulság kettős. Egyrészt a kiadott szótárban is találtam hibákat. A másik, hogy – mivel itt többnyire helyes szavak vannak felsorolva – a MS új ellenőrzője gyakori jó szavakat sem mindig ismer fel.

6. Szubjektív kiértékelés

A szubjektív kiértékelés a kereszteszt alapján objektív módon kinyert szóalakok vizsgálatából származik.

- Helyeske: Toldalékolása a ragok és jelek esetén a legpontosabb. A képzőknél kicsit túlgenerál. Korlátlan számú képzőt elfogad, (*legeslegellovasíthatatlanítottabbak*), és olyan toldalékokat is kezel, melyeket mások egyáltalán nem (*zsákosdi*). Az igeneves összetételekkel (*macskafogó, padlófeltörlés...*) nincs baj, a számnevek is pontosak, de egyéb összetételt ritkán enged meg. Kötőjeles összetétele szabad. Tiltó szabályok nincsenek. A betűn, számjegyen kívüli karaktereket nem kezeli. (*§-ának, °C, %-ot...*) A tulajdonnevek kisbetűsítését (pl. *-i* képző) algoritmikusan elvégzi. A forrásleírása a legtömörebb.
- HUNSPELL: Akad pontatlanul osztályozott szó. Szóösszetétele engedékeny, de legalább nem mond ellent az általános nyelvi szabályoknak. Szókészlete elég jó. Ez kezeli egyedül megkülönböztetően a rövid és a hosszú kötőjeleket. Van lehetőség tiltó szabályok alkalmazására, ezért elvileg még sokat javulhatna – ha lenne egy metaszintje a leírásoknak. A 6-3-as szabály ugyan nincs benne, de ritkán téved. A tulajdonnevek kisbetűsítését (pl. *-i* képző) algoritmikusan elvégzi. Jelenleg csak ez engedi meg felhasználói szótárában a ragozható tételek felvételét. Adatbázisa súrolja a kezelhető méret határát – mintegy 150 000 tétel.
- Office XP: Szókészlete elég jó. Van pár hiba a toldalékolásban – lelke még a régi 16 bites, ahol korlátok voltak a leírás összetettségére. Szóösszetétele elfogadható – talán a betűvel írt számok körül lehetnek nagyobb gubancok. Sok betűn és számon kívüli szót is jól kezel. Már nem fejlődik. Nem is érdemes, mert van jobb helyette. Adatbázisa súrolja a kezelhető méret határát – mintegy 150 000 tétel.
- Office 2002: A toldalékolása elég pontos, és szóösszetételben a legpontosabb. A tiltó szabályok hatékonyak. Sok betűn és számjegyen kívüli szót is jól kezel. Létezik metaleírás. A legjobban karbantartható. A kereszttesztek alapján legtöbbször ennek volt igaza a vitatott szóalakoknál. A 6-3-as szabályt már algoritmikusan kezeli. A tulajdonnevek kisbetűsítését szótári bejegyzésekkel oldja meg. A számok kezelése majdnem tökéletes. A felhasználói szótárban nincs lehetőség ragozható alakok felvételére. [10]Adatbázisa kezelhető méretű – mintegy 60 000 tétel.
- Office 2016: Egyedül a lefedettségi paraméterei jobbak a többinél, de ennek nagy az ára. A módszert nem ismerem, hogyan készült, de zsákutcának tűnik. Több sebből vérzik, és tulajdonképpen mindenben lemarad a többitől.
- Lektor: Látszik, hogy rég nem fejlődött, nem bővült. Én az 1993-as adatokkal dolgoztam. Főként tulajdonnevekből van hiánya, de egy-két gyakori köznév is hiányzik. Szóösszetételben nem erős. Toldalékolása precíz, kicsit konzervatív.

7. Összefoglaló

Az ellenőrzők mind hasznosak, de ez ma már nem elég. A minőség három összetevője: alapszókészlet, toldalékolási pontosság, szóösszetételek kezelése. Az ellenőrzők mindegyike valamiben erősebb a többinél, kivéve a legújabb MS szpellere. Kezdetekben a toldalékolásokon volt a fő hangsúly. A ragok, jelek használatára pontos leírások léteznek, de magyarban nem lehet csupán felszíni szabályok alapján osztályozni a szavakat. Ezzel kapcsolatos, hogy forrásleírása tömör legyen,

és lehetőleg ne lépje túl a 100 000-es tételszámot. Míg a HUNSPELL szóosztályozásának algoritmusa statisztikai eszközökre is támaszkodik[11], a Helyes-e szóbővítésénél mintaalapú az automatikus osztályozás módszere. Egyik sem kerülheti el az utólagos emberi felülvizsgálatot. Valószínűleg a neuronhálózatos megoldások sem eredményeznek jó megoldást, de ezt tudtommal még senki nem próbálta ki a magyarra, hacsak a Microsoft vagy a Google nem tette.
Ma a sarkalatos probléma a szóösszetételek kezelése. A kifinomult összetételkezelés érdekében szükség lenne pontosabb szabályrendszerre, amit az elemzők használnának. Addig is statisztikák segíthetnek, de a lehetséges szóösszetételek száma meghaladja azt a mértéket, amivel a statisztikai módszer elbír.
A lefedettség növelése nem kritikus. Persze szakszövegeknél fontos lenne kiegészítő szótárakra, amire volt is példa (orvosi, katonai Helyes-e?). Bővíteni lehet a szótárat, de inkább a toldaléktárakat kéne javítani, pontosítani. Minden bővítésnél figyelembe kell venni a 4.2 képletet a 2-es típusú hiba elkerülése érdekében. Ahol van kifinomult tiltó szabály, ott nagyobb esély van a javulásra.

8. Utóirat

- Nem teszteltem a Google tisztán valószínűségre alapozó, esetleg neuronhálózatos megoldását, annyira gyengének mutatkozik.
- Megszülettek az újított nyelvtant figyelembe vevő ellenőrzők.
- Fél év alatt a Microsoft másodlagos hibáinak száma harmadára csökkent. A becslésem szerint tíz éven belül elérik az elfogadható szintet.

Hivatkozások

1. Dömötör Andrea: HELYESÍRÁS-ELLENŐRZŐ PROGRAMOK VERSENYE
`http://anyanyelvapolo.hu/helyesiras-ellenorzo-programok-versenye/`
2. ORIGO: Szövegszerkesztők helyesírásversenye
`http://www.origo.hu/techbazis/szamitogep/20080923-megvizsgaltuk-a-helyesirasellenorzoket-microsoft-office-vs-openoffice.html`
3. Naszódi Mátyás: Statisztika megbízhatósága a nyelvészetben
Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015) Szeged, 2015
4. András Kornai: Frequency in morphology
In I. Kenesei (ed): *Approaches to Hungarian* Vol 4 (1992) 246-268
5. Németh László: A Szószablya fejlesztés
Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003) Szeged, 2003
6. MTA, Nyelvtudományi Intézet: Helyesírási tanácsadó
`http://xn-helyesrs-fza2j.mta.hu/helyesiras`
7. WEB: helyesírás http://www.magyarhelyesiras.hu/
8. webforditas.hu: Fordítási és helyesírási szolgáltatás
`http://www.webforditas.hu/helyesiras`
9. KISS G. Gábor: Magyar helyesírás-ellenőrző programok ellenőrzése és összehasonlítása *Könyv Papp Ferencnek* Debrecen KLTE (1991) 325–333.
10. Novák Attola: emMorph http://e-magyar.hu/hu/textmodules/emmorph
11. Halácsy P., Kornai A., Németh L., Rung A., Szakadát I. és Trón V.: A szógyakoriság és helyesírás-ellenőrzés
In: I. Kenesei (ed): *Approaches to Hungarian* Vol 4 (1992) 246-268

Szóbeágyazási modellek vizualizációjára és böngészésére szolgáló webes felület

Novák Attila[1,2], Siklósi Borbála[2], Wenszky Nóra[1]

[1] MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport ,
[2] Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar ,
1083 Budapest, Práter utca 50/a
e-mail:{novak.attila.siklosi.borbala}@itk.ppke.hu

Kivonat Cikkünkben egy word2vec alapú szóbeágyazási modellek vizualizációjára és böngészésére szolgáló webes felületet mutatunk be, amelybe a modellek lekérdezésén és vizualizációján túl számos komplex funkciót integráltunk. A webes felületen keresztül elérhető funkciók nagy méretű magyar és angol nyelvű korpuszból épített szóbeágyazási modelljeinkre épülnek.

1. Bevezetés

A szavak reprezentációjának meghatározása a nyelvtechnológiai alkalmazások számára alapvető feladat. A kérdés az, hogy milyen reprezentáció az, ami a szavak jelentését, vagy azok morfoszintaktikai, szintaktikai viselkedését is meg tudja ragadni. Angol nyelvre egyre népszerűbb a kézzel gyártott szimbolikus lexikai erőforrások és a nyers szövegből tanulható ritka diszkrét reprezentációk helyett a folytonos vektorreprezentációk alkalmazása, melyek hatékonyságát a neurális hálózatokra alapuló implementációk használatával több tanulmány is alátámasztotta [3,5,1,7]. Ezekben a kísérletekben és alkalmazásokban azonban a leírt módszereket általában a magyarhoz képest jóval kevesebb szóalakváltozattal operáló, kötött szórendű angol nyelvre alkalmazzák. Korábban megmutattuk, hogy az összetett morfológiájú nyelvek esetén is jó minőségű szóbeágyazási modell hozható létre a tanítókorpuszra alkalmazott megfelelő előfeldolgozás (a szavak külön tő- és morfológiaicímke-tokenekre bontása) esetén [8,9].[3]

A beágyazási modellek kiértékelésének egyik módszere az angol nyelvű modellek esetén az analógiatesztek elvégzése [4]. Ezeknél a teszteknél egy szópárosból és egy tesztszóból indulnak ki. A rendszer feladata annak a szónak a megtalálása, ami a tesztszóhoz az eredeti szópáros közötti relációnak megfelelően viszonyul. Például a *férfi – nő* páros és a *király* tesztszó esetén a várt eredmény a *királynő*. Elvégeztünk ugyan néhány ilyen tesztet, azonban mivel a többértelmű szavakhoz egy reprezentációs vektor tartozik, ezért a szópárok közötti relációkat kevésbé sikerült jól modellezni. Az előbbi példában a *nő* szó igei és főnévi jelentései keverednek, ezért a *férfi* és a *nő* szavak közötti távolság nem felel meg a *király* és a *királynő közötti távolságnak* (aminek oka a *király* szó többértelműsége is).

[3] A nyers szövegekből építettnél jobb.

Így csupán elvétve találtunk olyan analógiapéldákat, melyek helyes eredményt adtak.

A kvantitatív kiértékelés nehézsége ellenére is szerettük volna megvizsgálni a különböző módon létrehozott beágyazási modellek minőségét, illetve szimbolikus szemantikai tudást kinyerni belőlük. Ehhez létrehoztunk egy olyan webes felületet, aminek segítségével a modellek tartalma áttekinthető és könnyen kezelhető formában jelenik meg, illetve amibe további, a szóbeágyazási modellek értelmezhetőségét és felhasználhatóságát támogató megjelenítési formát és eszközt is integráltunk.

Cikkünkben a webes felület funkcióit és az eszköz segítségével feltárható jelenségeket mutatjuk be. A bemutatott demó a nagyközönség számára egyelőre nem érhető el, aminek oka elsősorban technikai, de a jövőben tervezzük a nyílt hozzáférés lehetőségét is megvalósítani.

2. Hasonló szavak lekérdezése

Egy szóbeágyazási modellben a lexikai elemek egy valós vektortér egyes pontjai, melyekben az egymáshoz szemantikailag és/vagy morfológiailag hasonló szavak egymáshoz közel, a jelentésben eltérő elemek egymástól távol esnek. Mindemellett, a vektoralgebrai műveletek is alkalmazhatók ebben a térben, tehát két elem szemantikai hasonlósága a két vektor távolságaként meghatározható, illetve a lexikai elemek pozícióját reprezentáló vektorok összege, azok jelentésbeli összegét határozzák meg [5,3].

Ezért a beágyazási modellek egyik alapvető funkciója egy tetszőleges szóhoz a modellben legközelebb elhelyezkedő szavak meghatározása a szavakat reprezentáló vektorok koszinusz távolsága alapján. A webes felületen lehetőség van arra, hogy egy tetszőleges szót beírva lekérdezzük a közelében található tetszőleges számú szót egy adott modellben. Jelenleg a felületen több modellt is tesztelni lehet:

- Felszíni szóalakokat tartalmazó modell (hu.surf): a nyers korpuszból tokenizálás után tanított modell, amiben a szavak toldalékolt alakja szerepel (magyar nyelvű modell).
- Tövesített alakokat tartalmazó modell (hu.ana): a modell építése előtt szófaji egyértelműsítést és lemmatizálást alkalmaztunk a korpuszra, majd a morfoszintaktikai információkat külön tokenként, a szótövek kontextusaként tartottuk meg (magyar nyelvű modell). Ilyen modellt tudomásunk szerint elsőként mi készítettünk: [8,9].
- Szófaji egyértelműsített modell (hu.pos): az előző modellhez hasonló, azzal a különbséggel, hogy a fő szófajcímkéket a szótövekre ragasztva tartottuk meg, így az azonos alakú, de más szófajú szavakhoz külön reprezentációt rendelt a modell (magyar nyelvű modell). Szófajcímkékkel annotált korpuszból angol nyelvre készítettek már beágyazási modelleket [10], de sem tövesítést, sem a ragok külön tokenekként való ábrázolását korábban nem alkalmazták.

– Angol nyelvű Wikipédia modell (wikien.pos): az angol Wikipédiából az előző modellnek megfelelő előfeldolgozással létrehozott modell (angol nyelvű modell). Itt is alkalmaztunk tövesítést is az előfeldolgozásnál, a ragokat külön token reprezentálja.
– Szemészeti kifejezéseket tartalmazó modell (szem.ana): az eredeti korpuszból épített magyar modell leszűkítése egy szemészeti korpusz szókincsére (magyar nyelvű modell)
– Lexikai erőforrások szemantikai kategóriáit tartalmazó modellek (4lang, ldocehu, rogethu): három angol nyelvű lexikai erőforrásból (4lang, Longman Dictionary of Contemporary English, Roget's Thesaurus) épített modell [6], mindegyikben az egyes kategóriákhoz felsorolt példaszavakat azok vektorainak átlagolásával és átlagvektor magyar vektortérbe vetítésével létrejött reprezentációval ábrázoljuk (angol nyelvű címkéket a magyar nyelvű vektortérben megjelenítő modellek). Ezeket a modelleket használjuk az 5. és a 6. részben említett funkciók megvalósításánál.

A modellek létrehozásának részletei a következő cikkekben olvashatók: [8,6].

Az egyes lekérdezésekre kapott válaszban a kérdőszóhoz mért távolság alapján rendezve jelenik meg a szólista, amiben szerepel az egyes elemek korpuszbeli gyakorisága és a hasonlóság mértéke (koszinusztávolság) is. Az 1. ábrán két lekérdezés eredménye látható. Az első listában az elemzett hu.ana modellből kérdeztük le az *alma* szóhoz legközelebb eső 10 szót, míg a második listán a nyers korpuszból létrehozott hu.surf modellből a *kenyerek* szóalakhoz tartozó 10 leghasonlóbb szóalak látható. A második esetben a hasonlóság nem csak szemantikai, hanem morfoszintaktikai vonatkozásban is teljesül (a korpuszban ritkábban előforduló szavak esetén azonban az utóbbi modell kevésbé jól használható).

0	alma	1	63906
1	körte	0.8392	13339
2	eper	0.8356	16159
3	banán	0.8222	17732
4	szilva	0.8046	12602
5	őszibarack	0.8011	4698
6	uborka	0.7971	14735
7	répa	0.7937	14107
8	cseresznye	0.7848	11676
9	ananász	0.7820	4827
10	dinnye	0.7689	11428

0	kenyerek	1	2270
1	zsemlék	0.8105	283
2	péksütemények	0.8048	997
3	kekszek	0.7972	1046
4	pékáruk	0.7957	771
5	tészták	0.7881	2466
6	lepények	0.7849	202
7	kiflik	0.7843	349
8	kalácsok	0.7841	277
9	sonkák	0.7836	613
10	pogácsák	0.7787	539

1. ábra. Példa hasonló szavak lekérdezésének eredményére a tövesített és a felszíni alakokat tartalmazó modellből

A lekérdezésekre kapott listák elemeit interaktív módon (egérkattintással) is kiválaszthatjuk, ekkor a kattintott szóhoz legközelebbi elemek listáját is megkapjuk a beállított modellből. A lexikai erőforrások szemantikai kategóriáit tartalmazó modellek (4lang, ldocehu, rogethu) kiválasztása esetén a rendszer magyar szavak beírásakor a vektortérben az adott modellben legközelebbi címkéket adja vissza, azokra kattintva pedig fordítva az adott címkéhez legközelebbi szavak jelennek meg.

3. Klaszterezés és vizualizáció

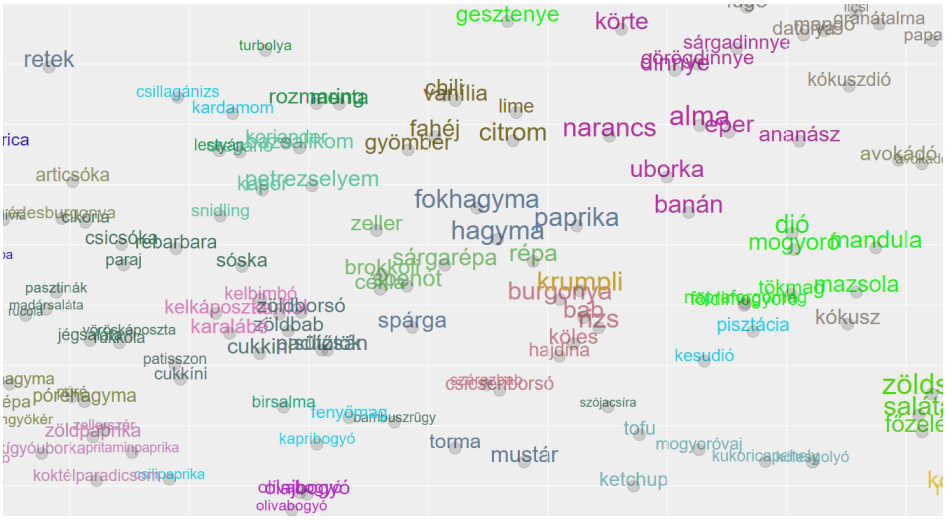
Ha egy szólistában szereplő szavak egymáshoz való viszonyát szeretnénk megjeleníteni, akkor a felület lehetőséget ad arra is, hogy egy listát megadva, az abban szereplő szavakhoz a kiválasztott modell által rendelt vektorok alapján azokat klaszterezve, csoportosítva jelenítsük meg. Így az egymáshoz közel álló szavak azonos, míg a távolabbi szavak külön klaszterben jelennek meg (az elkülönítés érzékenysége állítható paraméter). Az alkalmazott algoritmus részleteit lásd: [8]. A klaszterezésre szánt listát természetesen úgy is előállíthatjuk, hogy az előző pontban ismertetett módon egy kiinduló szóból elindulva az annak a közelében található szavak listáját (vagy akár több ilyen listát egyesítve) csoportosítjuk az eredményt. Ezzel a módszerrel könnyen kiszűrhetjük az esetleg zajként megjelenő találatokat, vagy egy hosszabb listát szemantikailag releváns alcsoportokra bonthatunk.

A fogalmakat reprezentáló vektorok egy szemantikai térben helyezik el az egyes lexikai elemeket, így ez a szerveződés látványosan vizualizálható. Ehhez a listában szereplő szavakhoz tartozó sokdimenziós vektorokat egy kétdimenziós térbe képeztük le a t-sne algoritmus alkalmazásával [2]. A módszer lényege, hogy a szavak sokdimenziós térben való páronkénti távolságának megfelelő eloszlást közelítve helyezi el azokat a kétdimenziós térben, megtartva tehát az elemek közötti távolságok eredeti arányát. Így könnyen áttekinthetővé válik a szavak szerveződése, a jelentésbeli különbségek jól követhetőek és felmérhetőek.

A vizualizáció során a klaszterezés eredményeit is megjelenítettük, a különböző klaszterbe került szavakat különböző színnel jelenítve meg. Az így létrejött ábrán jól követhetővé váltak a klaszterek közötti távolságok is. A 2. ábra egy ilyen leképezés részletét ábrázolja.

4. Analóg szókapcsolatok megjelenítése

Bár a felületen jelenleg kipróbálható modellekben csupán szavak reprezentációja található meg (természetesen a felület összes funkciója alkalmas többszavas kifejezéseket tartalmazó modellek kezelésére is), néhány olyan lekérdezésre is lehetőség van, ahol több szót együttesen vizsgálhatunk. Egyrészt az egyszavas lekérdezéshez hasonlóan több szót is megadhatunk a lekérdező mezőben, ekkor az egyes szavakhoz tartozó beágyazási vektorokat összeadjuk és az összegvektor-



2. ábra. Gyümölcsök és zöldségek

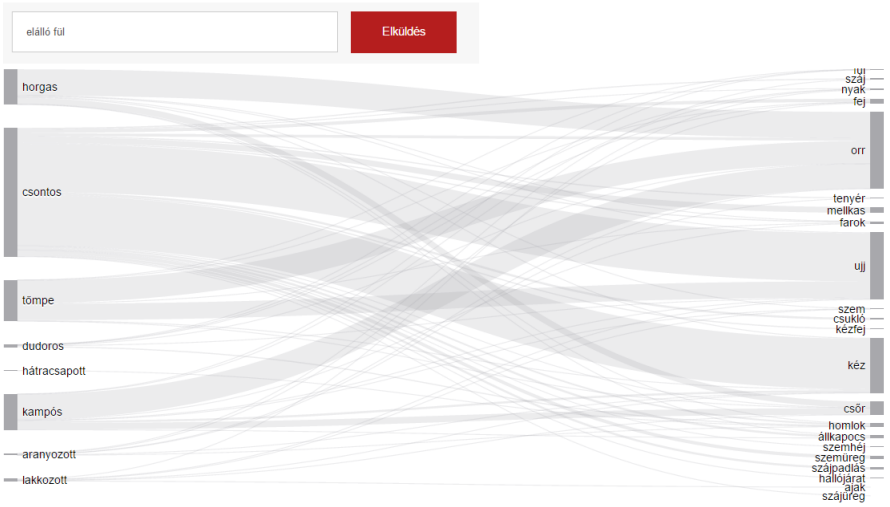
hoz legközelebbi szavakat jelenítjük meg az adott modellből.[4] A lekérdezésben algebrai műveletek is megadhatók (pl. a pozitív és negatív előjellel összegzendő szavak halmaza), így a felület a bevezetésben említett analógiák lekérdezésére is alkalmas. Megszorítást tehetünk arra is, hogy a rendszer a modellből csak egy adott szóhalmaz elemei közül a leghasonlóbbakat adja vissza.

Egy másik funkció használatakor pedig egy több szóból álló kifejezés lekérdezése során mindegyik szóhoz lekérdezzük az annak környékén található hasonló szavakat, majd az összes kombináció szerint párba rendezzük őket az eredeti szórend szerint és az így kapott bigramok korpuszbeli gyakoriságát reprezentáló ábrán jelenítjük meg az eredményt. A 3. ábrán látható *elálló fül* kifejezés esetén tehát az *elálló* szóalakhoz legközelebbi szavak a *horgas, csontos, tömpe, dudoros, hátracsapott, kampós, aranyozott, lakkozott*, tehát az ábra bal oszlopában látható szavak, a *fül* szóhoz hasonlók pedig a jobb oldalon szereplő *száj, nyak, fej, orr, tenyér, mellkas, farok, ujj, szem, csukló, kézfej, kéz, csőr, homlok, stb.*. A két oszlop közötti összeköttetések vastagsága jelöli az összekötött bigramok gyakoriságát. Így látható, hogy az eredeti *elálló fül* kifejezéshez hasonló bigramokat kaptunk, mint pl. *horgas orr, tömpe orr, csontos kéz, stb.*

5. A Dologfelismerő

Egy másik cikkünkben [6] olyan módszert mutatunk be (Dologfelismerő néven), amely egy szóbeágyazási modellhez az eredeti modell értelmezését segítő szemantikai kategóriacímkékkel összerendelt vektorokat ad hozzá. A hozzáadott címké-

[4] Mivel koszinusz-távolságot használunk metrikaként, a vektorösszeghez és az átlagvektorhoz képest számított távolság azonos, lévén ezek iránya azonos.

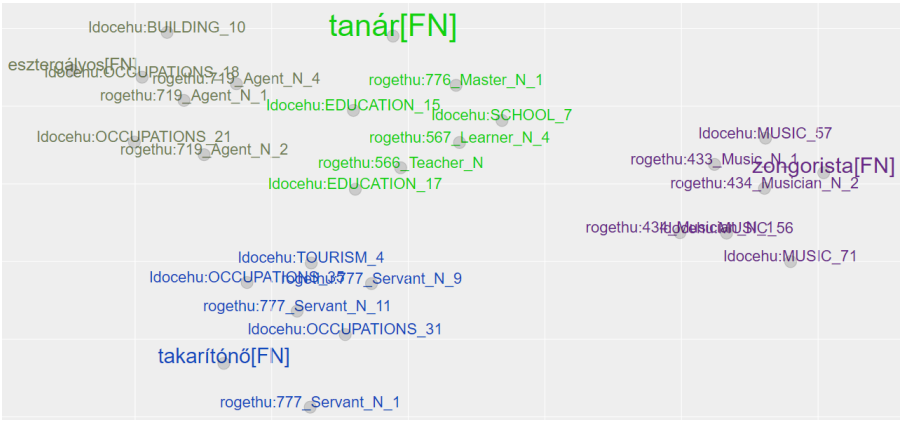


3. ábra. Az *elálló fülhöz* hasonló kifejezések

ket létező lexikai erőforrásokból, azok automatikus transzformációjával illesztjük az eredeti beágyazási térbe. Ennek köszönhetően az eredetileg nagyon sok szót tartalmazó szemantikai térben jóval kisebb számú referenciapontot helyezünk el, és a lekérdezésnél csak ezeket használjuk, ami a modellt átláthatóbbá teszi. A bemutatott algoritmus az eredeti korpuszban lévő összes szóhoz képes kategóriacímkét rendelni, függetlenül attól, hogy az adott szóalak a címkék létrehozásához használt lexikai erőforrásban szerepelt-e. Továbbá, a módszer nyelvfüggetlen, a felhasznált erőforrások nyelve (itt angol volt) nem szükségszerűen azonos az eredeti szóbeágyazási modell (itt magyar) nyelvével. Ezzel a módszerrel jöttek létre a 4lang, ldocehu és rogethu modellek.

A webes felületbe is integráltuk ezt a funkciót. Egyrészt lehetőség van a különböző kategorizációs modellekből (4lang, rogethu, ldocehu) egy tetszőleges szóhoz az ahhoz rendelt kategóriacímkéket lekérdezni. Mivel a hozzárendelés során egy köztes lépésben egy angol modellt is felhasználunk, ezért a nyelvek közötti transzformációt is be tudjuk mutatni oly módon, hogy a lekérdezett szóhoz az angol modellben (wikien.pos) legközelebb álló szavakat jelenítjük meg. Az eredmények, akár az angol kapcsolódó szavak, akár a kategóriacímkék esetén, itt is egy-egy, a hasonlóság mértéke szerint rendezett listában jelennek meg.

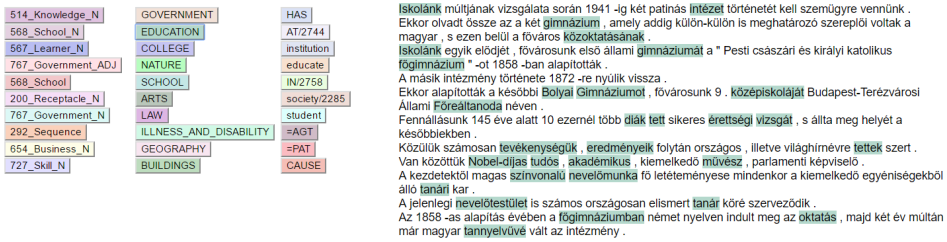
Emellett a kategóriacímkéket a fent bemutatott kétdimenziós ábrán is meg tudjuk jeleníteni. Tetszőleges számú szóhoz tetszőleges címkemodellekből tetszőleges számú címkét azonos szemantikai térbe való transzformáció után egyetlen ábrán jelenítünk meg. Ezáltal az eredeti beágyazási tér egyes területeit a Dologfelismerőben definiált kategóriacímkékkel vizuálisan is annotálni tudjuk. A 4. ábrán négy szóhoz (*zongorista, tanár, esztergályos, takarítónő*) két modellből 3-3 hozzárendelt címke és ezek elhelyezkedése látható a 2 dimenziós térbe leképezve.



4. ábra. Foglalkozások és címkéik

6. Szövegcímkéző

A felületre egy olyan funkciót is integráltunk, ami a Dologfelismerő címkéző funkcióját az ott használt modellekre támaszkodva hosszabb szövegekre valósítja meg (némileg hasonlóan a Wikifierhez[5]). A beadott szöveget elemezzük (lemmatizálás, PoS-taggelés) és a tartalmas szavakhoz (jellemzően főnevek, igék, melléknevek) lekérdezzük az 5 legközelebb álló kategóriacímkét mindhárom modellből, majd ezeket összesítjük a teljes szövegre nézve és gyakoriság szerinti sorrendben jelenítjük meg az eredeti szöveg mellett. Ez a funkció a szöveg tematikus besorolását, egyfajta szövegkivonatolást valósít meg. A megjelenített címkék ebben az esetben interaktív gombként is működnek, egy-egy címke kiválasztásakor a szövegben az olyan címkét kapott szavak kiemelődnek. A 5. ábrán egy ilyen módon felcímkézett szöveg látható.



5. ábra. A „Thingifájer”

[5] http://wikifier.org/

7. Konklúzió

A szóbeágyazás a szavak jelentésének ábrázolására hatékonyan használható reprezentációs módszer, azonban a létrejött modellek minőségének ellenőrzése nehéz feladat nem csak azért, mert a modellek közvetlen kiértékelésére alkalmazható kvantitatív módszerek nyelvenkénti adaptációja nehéz, hanem azért is, mert a szemantikai reprezentáció minőségének meghatározása szubjektív. A bemutatott webes felülettel célunk az volt, hogy a különböző módon létrehozott magyar nyelvű szóbeágyazási modelleket vizsgálni tudjuk, többféle módon jelenítve meg az általuk definiált szemantikai teret. Emellett a felület nagyon hatékonyan használható különböző szemantikai osztályozási, lexikográfiai feladatok elvégzésére. A szóbeágyazási modellek felhasználásával megvalósított komplexebb algoritmusaink egy része szintén elérhető a webes felületről.

Hivatkozások

1. Baroni, M., Dinu, G., Kruszewski, G.: Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 238–247. Association for Computational Linguistics, Baltimore, Maryland (June 2014)
2. van der Maaten, L., Hinton, G.: Visualizing high-dimensional data using t-sne (2008)
3. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
4. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States. pp. 3111–3119 (2013)
5. Mikolov, T., Yih, W., Zweig, G.: Linguistic regularities in continuous space word representations. In: Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA. pp. 746–751 (2013)
6. Novák, A., Siklósi, B.: A dologfelismerő. XIII. Magyar Számítógépes Nyelvészeti Konferencia (2017)
7. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014)
8. Siklósi, B., Novák, A.: Beágyazási modellek alkalmazása lexikai kategorizációs feladatokra. XII. Magyar Számítógépes Nyelvészeti Konferencia pp. 3–14 (2016)
9. Siklósi, B.: Using embedding models for lexical categorization in morphologically rich languages. In: Gelbukh, A. (ed.) Computational Linguistics and Intelligent Text Processing: 17th International Conference, CICLing 2016. Springer International Publishing, Cham., Konya, Turkey (April 2016)
10. Trask, A., Michalak, P., Liu, J.: sense2vec - A fast and accurate method for word sense disambiguation in neural word embeddings. CoRR abs/1511.06388 (2015), http://arxiv.org/abs/1511.06388

Függőségi elemzésen alapuló magyar nyelvű keresőrendszer

Zsibrita János[1], Farkas Richárd[1], Vincze Veronika[1,2]

[1]Szegedi Tudományegyetem, Informatikai Intézet
[2]MTA-SZTE Mesterséges Intelligencia Kutatócsoport
{zsibrita,rfarkas,vinczev}@inf.u-szeged.hu

Kivonat A cikkben bemutatjuk webes keresőrendszerünket, mely függőségi elemzésen alapuló kereséseket tesz lehetővé magyar nyelvű szövegekben. A rendszer azokat a szókapcsolatokat adja vissza, ahol a keresett szó és annak bővítménye a keresőkifejezésben megadott nyelvtani viszonyban állnak egymással. Van mód a találatok szűkítésére is morfológiai, illetve szótőre vonatkozó jellemzőkre építve. A rendszer alapjait a magyarlanc morfológiai és szintaktikai elemző moduljai jelentik.
Kulcsszavak: keresés, szintaxis, morfológia, információkinyerés

1. Bevezetés

Az információkinyerés és -feldolgozás egyik fontos lépése a szövegek nyelvi előfeldolgozása, azaz a szövegek mondatokra, majd szavakra bontása, morfológiai elemzése és szófaji egyértelműsítése, illetve mély szintaktikai elemzése. A mély nyelvi jellegű adatok felhasználására épülő rendszerekkel általában pontosabb eredményeket kaphatunk, mint a felszíni jegyekre, puszta szóalakokra vagy szótövekre támaszkodó alkalmazások. Ezért a mély nyelvi elemzések kiaknázása igen hasznosnak bizonyulhat a nyelvtechnológiai alkalmazások terén, különösen a keresésen alapuló módszerek esetében.

Ebben a cikkben bemutatjuk függőségi elemzésre épülő keresőrendszerünket, melynek segítségével magyar nyelvű szövegekből nyerhetjük ki azokat a szókapcsolatokat, melyek a keresőkifejezésben meghatározott nyelvtani viszonyban állnak egymással. Tudomásunk szerint ez az első olyan, magyar nyelvű keresőrendszer, mely nagyméretű szöveges adatbázisokban képes szintaktikai alapú keresést végrehajtani, a szótövek kötelező meghatározása nélkül. A rendszerben lehetőség nyílik a találatok morfológiai alapon történő szűkítésére is. A továbbiakban részletesen bemutatjuk a keresőrendszert, majd példákkal illusztráljuk működését.

2. Kapcsolódó irodalom

A Nyelvtudományi Intézet gondozásában működő Nemzeti Korpuszportál[1] [1] felsorolja azokat a magyar nyelvű korpuszokat, amelyekhez rendelkezésre áll on-

[1] http://corpus.nytud.hu/nkp/

line kereső. Ezek közül most – mint általános szövegekben való keresőeszközöket – a Magyar Nemzeti Szövegtár [2] keresőjét és a Mazsola nevű eszközt [3] tekintjük át részletesebben.

A Magyar Nemzeti Szövegtár 1.0 változata [2] kb. 200 millió, 2.0 változata [4,5] közel egymilliárd szót tartalmaz. Mindkét változatában megtaláljuk minden egyes szó lemmáját és morfológiai elemzését, melyeket a keresésben is tudunk hasznosítani. A találatok konkordancia formában jelennek meg. Szókapcsolatokra is lehetséges keresni a keresett szó szűkebb környezetében előforduló más szavak vagy azok morfológiai jellemzőinek meghatározásával.

A Mazsola nevű eszközzel [3] a magyar igék bővítményszerkezetének feltérképezése válik lehetségessé. Szintén a Magyar Nemzeti Szövegtár [2] szövegeiben képes keresni, a szövegek morfológiai elemzését felhasználva. Segítségével megjeleníthető, hogy egy adott ige mellett milyen bővítmények jelenhetnek meg. Esetragok és névutók alapján, illetve a bővítmény szótöve alapján is lehetséges keresni az adatbázisban, majd a találatok gyakoriság szerint rangsorolva jelennek meg. A keresés egysége a mondat, azaz az igével egy mondatban szereplő bővítményeket ad vissza a kereső (melyek nem szükségszerűen szintaktikai vonzatai az adott igének).

A Mazsola mellett a jelenlegi munkában bemutatott keresőrendszerhez legközelebb a Szegedi Tudományegyetemen korábban kifejlesztett néprajzi kereső áll [6]. A kereső célja, hogy különféle néprajzi dokumentumokban egész mondatos kereséseket hajtson végre, azaz olyan dokumentumokat ad vissza, ahol a keresett ige és vonzatai a keresőkifejezésben megadott szintaktikai viszonyban állnak egymással. A háttéradatbázis magyar nyelvű hiedelmeket, táltosszövegeket, illetve meséket tartalmaz, összesen kb. 750 ezer szövegszóból áll. A keresőmondat függőségi elemzésére épülve a megtalált grammatikai relációknak és szótöveknek megfelelő illeszkedéseket keres a rendszer a szövegekben, morfológiai alapú, illetve szótövektől független keresésre azonban nem nyílik lehetőség.

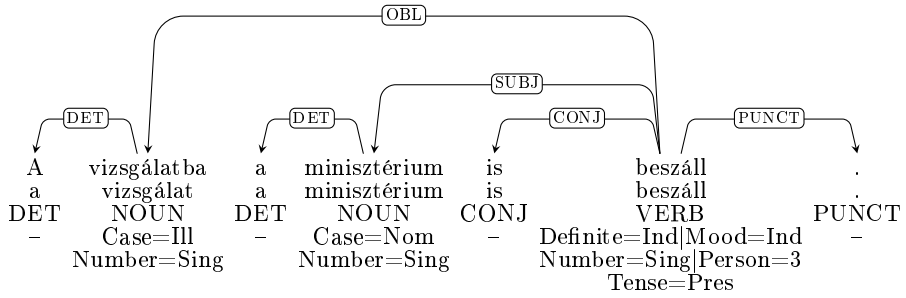
3. A keresőrendszer

Ebben a részben áttekintjük a keresőrendszer működési alapjait, illetve bemutatjuk röviden a mögöttes adatbázist.

3.1. A keresőrendszer működése

A magyar nyelv diskurzuskonfigurációs nyelv, azaz a bővítmények mondatbeli (szintaktikai) szerepére a szórend nincs hatással: a szintaktikailag összetartozó elemek nem feltétlenül szomszédosak, hanem előfordulhatnak egymástól távol is a mondaton belül. Ebből következően a magyar nyelvre nem optimálisak azok a keresési stratégiák, melyek pusztán a szövegkörnyezetet, azaz a keresett szó közvetlen környezetében található elemeket veszik figyelembe, szükség van szintaktikai információkra is.

A Javában implementált keresőrendszer az Elasticsearch[2] nyílt hozzáférésű eszközre épül. A keresések alapjául morfológiailag és szintaktikailag elemzett szövegek szolgálnak (lásd 3.2. rész). A morfológiai elemzés az univerzális morfológia magyarra adaptált elveinek [7] felel meg, míg a függőségi elemzésben a Szeged Dependencia Treebank [8] elveit követjük. Egy szövegszóhoz rendelkezésünkre áll annak lemmája, szófaja és részletes morfológiai elemzése, valamint a mondatbeli nyelvtani szerepe és az, hogy minek a bővítménye (azaz mi a szülő csomópontja a függőségi fában), lásd 1. ábra. A keresés során mindezen információt képesek vagyunk hasznosítani.



1. ábra: Morfológiai és szintaktikai annotáció.

A keresőrendszer alapvetően függőségi viszonyokra épül, azaz olyan szópárokat ad vissza találatként, amelyek között az adott szintaktikai reláció található (pl. alany-igei állítmány párok). A keresés során kötelező megadni a keresett függőségi viszonyt, továbbá lehetőség van a keresés szűkítésére más információk megadásával: mind a szülő, mind a gyermek csomópont esetében lehetséges azok lemmáját és/vagy szófaját, morfológiai jegyeit meghatározni.

Találatként olyan szópárokat kapunk vissza, amelyek között a megadott szintaktikai viszony szerepel, illetve minden további (szófajra, morfológiára, illetve lemmára vonatkozó) feltétel fennáll.

3.2. Az adatbázis

A keresőrendszer jelenleg két forrásból származó szövegekből képes visszaadni a meghatározott nyelvtani viszonyban álló szövegrészeket. Az egyik forrás a teljes Szeged Dependencia Treebank [8], melyben kézzel annotált morfológiai és szintaktikai elemzéseket találhatunk. Második forrásként az index.hu hírportálról töltöttünk le híreket (összesen 50,000 cikk) 2016 októberében és novemberében, majd azokat automatikusan elemeztük a magyarlanc 3.0 elemző lánc [9] segítségével. Az így kapott morfológiai és függőségi elemzések szolgálnak a keresések alapjául. Lehetőség van a találatok szűkítésére aszerint is, hogy melyik

[2] http://www.elastic.co/products/elasticsearch

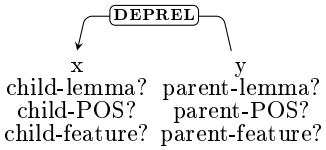
korpuszrészben szeretnénk keresni, így akár korpuszközi összehasonlításokat is végezhetünk.

A keresőrendszer mögött álló adatbázist folyamatosan bővítjük.

4. Keresési példák

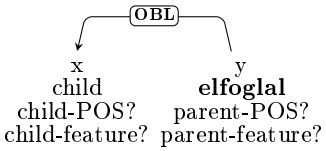
Az alábbiakban részletesebben is bemutatjuk, milyen keresési lehetőségeket biztosít a rendszer, továbbá arra is kitérünk, hogy a függőségi viszonyokon alapuló keresés milyen többletet jelent az egyszerű szó- vagy morfológiai alapú keresőkhöz képest.

A 2. ábrán láthatjuk a keresés sémáját. x és y jelöli a keresett szavakat (szóalakokat) – ezeket fogja visszaadni a keresőrendszer. Kötelező megadni a függőségi relációt (DEPREL), míg a kérdőjellel jelölt elemek opcionálisan megadhatók, akár egy, akár több elem is. Az alábbiakban ezekre mutatunk néhány példát.



2. ábra: A keresés sémája.

A szótövön alapuló kereséskor a keresett szó lemmáját kell megadnunk, illetve azt, hogy milyen szintaktikai viszonyt jelöl. Ha például arra vagyunk kíváncsiak, hogy milyen (nem alanyi, tárgyi és részeshatározói) vonzatai lehetnek az *elfoglal* igének, akkor a 3. ábrán látható keresést kell végrehajtanunk (vastaggal kiemelve a megadott elemeket):

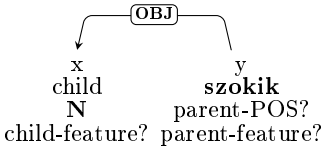


3. ábra: Keresés az *elfoglal* + OBL szerkezetre.

Lehetséges találatok például a következők:

pecsenyéjükkel/pecsenye@NOUN OBL elfoglalva/elfoglal@VERB
kihelyezésével/kihelyezés@NOUN OBL elfoglalva/elfoglal@VERB
Afganisztánban/Afganisztán@NOUN OBL elfoglalta/elfoglal@VERB

Egy másik példát nézve, a magyar *szokott* ige előfordulhat főigeként, illetve segédigeként is. Előbbi esetben jellemzően tárgyas igeként fordul elő, jelentése „hozzászokik", „megszok valamit", utóbbi esetben szokásos cselekvést jelöl, ilyenkor gyakran főnévi igenevet vonz, melynek szintén lehet tárgya. Ha arra vagyunk kíváncsiak, hogy mihez (milyen főnévhez) lehet hozzászokni, akkor az alábbi keresőkifejezést használhatjuk:



4. ábra: Keresés a *szokik* + OBJ szerkezetre.

A kapott találatok például a következők:

nagyapát/nagyapa@NOUN OBJ szoktam/szokik@VERB
világát/világ@NOUN OBJ szokták/szokik@VERB
vidéket/vidék@NOUN OBJ szokja/szokik@VERB

A fenti példák eredeti környezetéből kiderül, hogy valaminek/valakinek a megszokásáról esik szó éppen.

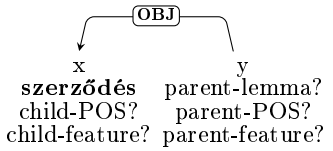
Ha nem áll rendelkezésre szintaktikai elemzés, akkor olyan keresőkifejezést írhatunk, ami a *szokik* ige közelében keres tárgyesetben álló főneveket. Ezzel téves találatokat is kaphatunk, például:

aranyórát/aranyóra@NOUN szokott/szokik@VERB

A fenti példát tartalmazó mondatból (*időnként el szokott lopni egy aranyórát*) jól látszik, hogy noha az *aranyórát* szó tényleg tárgyesetben áll, mégsem a *szokott* ige tárgya, hanem a *lopni* igéé, így nem állnak egymással közvetlen szintaktikai kapcsolatban.

A fentiek arra is rávilágítanak, hogy a szintaktikai információk felhasználásával pontosabb eredményeket kaphatunk, hiszen a fenti esetekben az igék és igenevek megfelelő vonzatait kapjuk meg, míg a pusztán szövegkörnyezetet (és morfológiát) felhasználó keresők a mondatban szereplő egyéb, nem az igenévhez tartozó vonzatokat is megjelenítenék.

A kereső arra is lehetőséget ad, hogy a vonzat oldaláról határozzuk meg a keresőkifejezést. Például ha arra vagyunk kíváncsiak, hogy a *szerződés* szó milyen szavaknak lehet a tárgya, akkor a következőképpen tehetjük meg:



5. ábra: Keresés a *szerződés* (OBJ) szerkezetre.

Lehetséges találatok a következők:

szerződést/szerződés@NOUN OBJ aláírták/aláír@VERB
Szerződést/szerződés@NOUN OBJ kötöttek/köt@VERB
szerződést/szerződés@NOUN OBJ írt/ír@VERB
szerződést/szerződés@NOUN OBJ bontott/bont@VERB
szerződést/szerződés@NOUN OBJ megtámadni/megtámad@VERB

A találatok gyakorisági mutatói arra is rámutatnak, hogy mik a magyar nyelvben gyakorta használatos szókapcsolatok, így akár a kollokációk vagy többszavas kifejezések megtalálásában és feltérképezésében is segítséget nyújthat a kereső.

5. Szintaktikai és szóalapú keresés

A magyar nyelvre eddig rendelkezésre álló keresők és lekérdezők többsége szóalapon működik, melyek a szóalakon kívül a szó lemmáját és morfológiai tulajdonságait képesek figyelembe venni a keresés során. Az általunk kifejlesztett rendszer több pontban is különbözik tőlük, melyeket az alábbiakban összegzünk:

- a keresés függőségi viszonyokon alapul, emellett a lemma és morfológiai információk is beépíthetők a keresőkifejezésbe,
- a bővítmény meghatározása is lehetséges, nem csak a szülő csomóponté,
- lexikális információ (a szóalak vagy szótő) meghatározása nélkül is tudunk keresni, csupán nyelvtani információk alapján,
- a keresés több találatot eredményez (nő a fedés), mivel a szintaktikai viszonyok figyelembevételével képes az egymástól távol eső, ám a lekérdezésnek megfelelő találatokat is visszaadni (pl. távoli függőségek),
- a keresés pontosabb találatokat eredményez (nő a pontosság), hiszen nem adja vissza azokat a szópárokat, amelyek lemmája és/vagy morfológiai elemzése megfelel a lekérdezésnek, ám egymással nem állnak szintaktikai kapcsolatban (pl. igenevek vonzatai).

6. Összegzés

A cikkben bemutattuk keresőrendszerünket, mely függőségi elemzésen alapuló kereséseket tesz lehetővé magyar nyelvű szövegekben. A rendszer azokat a

szókapcsolatokat adja vissza, ahol a keresett szó és annak bővítménye a keresőkifejezésben megadott nyelvtani viszonyban állnak egymással. A találatok pontosíthatók morfológiai, illetve szótőre vonatkozó jellemzők segítségével.

A szintaktikai alapokon nyugvó kereső felhasználhatósága többféle oldalról is jelentős. Egyrészt pontosabb találatokat ad, mint a pusztán szóalapon kereső rendszerek, így a magasabb rendű nyelvtechnológiai alkalmazások (pl. információkinyerés) is jobb eredményeket tudnak elérni. Másrészt korpusznyelvészeti és lexikológiai vizsgálatok során is lehetséges hasznosítani a keresőt. Továbbá akár a magyar nyelvtan, akár a magyar mint idegen nyelv oktatását is segítheti a rendszer.

A keresőrendszer mindenki által használható és szabadon elérhető a http://rgai.inf.u-szeged.hu/depsearch/ webcímen.

Köszönetnyilvánítás

Farkas Richárd kutatásait az MTA Bolyai János ösztöndíja támogatta.

Hivatkozások

1. Sass, B.: Nyelvészeti szövegkeresők, Nemzeti Korpuszportál. Magyar Tudomány **7** (2016) 798–808
2. Váradi, T.: The Hungarian National Corpus. In: Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002), Las Palmas de Gran Canaria, European Language Resources Association (2002) 385–389
3. Sass, B.: The Verb Argument Browser. In Horák, A., Kopeček, I., Pala, K., Sojka, P., eds.: Proceedings of the 11th International Conference on Text, Speech and Dialogue, Berlin, Heidelberg, Springer Verlag (2008) 187–192
4. Oravecz, Cs., Váradi, T., Sass, B.: The Hungarian Gigaword Corpus. In Chair), N.C.C., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S., eds.: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, European Language Resources Association (ELRA) (2014)
5. Oravecz, Cs., Sass, B., Váradi, T.: Mennyiségből minőséget. Nyelvtechnológiai kihívások és tanulságok az MNSz új változatának elkészítésében. In: XI. Magyar Számítógépes Nyelvészeti Konferencia, Szeged, Szegedi Tudományegyetem (2015) 109–121
6. Zsibrita, J., Vincze, V.: Magyar nyelvű néprajzi keresőrendszer. In: IX. Magyar Számítógépes Nyelvészeti Konferencia, Szeged, Szegedi Tudományegyetem (2013) 361–367
7. Vincze, V., Simkó, K.I., Szántó, Zs., Farkas, R.: Universal Dependencies and Morphology for Hungarian – and on the Price of Universality (2017) Elfogadva az EACL 2017 konferenciára.
8. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian Dependency Treebank. In: Proceedings of LREC 2010, Valletta, Malta, ELRA (2010)
9. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP. (2013) 763–771

VIII. Angol nyelvű absztraktok

State of the Hungarian Spell Checkers

Mátyás Naszódi, e-mail: naszodim@morphologic.hu

MorphoLogic, 1122 Ráth György utca 36. Hungary

Abstract The quality of spell checkers depends on the applied method of constructing and maintaining their databases. The size of the database may limit the achievable quality. The present article discusses the methodology of the objective evaluation of spell checkers and the theoretical limits of testing. It attempts to compare the available programs impartially, and to show the advantages of the applied methods used for the construction of linguistic databases. Finally, it reviews the directions of possible improvement.

Keywords: spell checker, statistics, linguistic tool, quality of spell checker

Syntactic Tagsets Affect Parsing Efficiency

Katalin Ilona Simkó[1,2], Viktória Kovács[2], Veronika Vincze[1,3]

[1]University of Szeged, Department of Informatics
Szeged, Árpád tér 2.
simko@hung.u-szeged.hu
[2]University of Szeged, Department of General Linguistics
Szeged, Egyetem u. 2.
viki921015@hotmail.com
[3]MTA-SZTE Research Group on Artificial Intelligence
Szeged, Tisza Lajos körút 103.
vinczev@inf.u-szeged.hu

Nowadays, the choice between different syntactic frameworks is getting bigger and bigger not only in theoretical, but in computational linguistics. For Hungarian, multiple representations are available for different syntactic frameworks.

In this paper, we present our findings on using different dependency labelsets in the syntactic analysis. We investigated the effect of different labelsets on the syntactic analysis itself and in applications using the syntactic parsing.

The study is based on Universal Dependencies for Hungarian [1] and in our investigation, we look at labels of adverbials, subordinating clauses and function words.

For evaluation of the syntactic parsing, we use labeled and unlabeled attachment scores as well as F-scores achieved on the content words. We believe that the syntactic parse itself is only a preprocessing step for other applications, so we try our representations (labelsets) in an NLP task classifying texts for mild cognitive impairment [2].

Our representations show improvement for syntactic parsing as well as significant improvement in the mild cognitive impairment task.

Our paper shows that choosing an appropriate syntactic representation has a powerful effect on the results of any syntax-dependent NLP application.

References

1. Vincze, V., Farkas, R., Simkó, K.I., Szántó, Zs., Varga, V.: Univerzális dependencia és morfológia magyar nyelvre. In: XII. Magyar Számítógépes Nyelvészeti Konferencia, Szeged (2015) 322–329
2. Vincze, V., Gosztolya, G., Tóth, L., Hoffmann, I., Szatlóczki, G., Bánréti, Z., Pákáski, M., Kálmán, J.: Detecting mild cognitive impairment by exploiting linguistic information from transcripts. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Berlin, Germany, Association for Computational Linguistics (2016) 181–187

Szerzői index, névmutató